A NEW SOLUTION AND CONCRETE IMPLEMENTATION STEPS THAT CAN REALIZE A TRULY UNIVERSAL ARTIFICIAL INTELLIGENCE

Chen Yongcong
New Millennium Future Technology Co., LTD.
Beijing
yongcongchen@sina.com

Zeng Ting
Tsinghua University
BeiJing
ceng-t@mail.tsinghua.edu.cn

Zhang Jun
Shenyang Normal University
Shenyang

ABSTRACT

At present, the mainstream artificial intelligence generally adopts the technical path of "attention mechanism + deep learning" + "reinforcement learning". It has made great progress in the field of AIGC (Artificial Intelligence Generated Content), setting off the technical wave of big models[2][13]. But in areas that need to interact with the actual environment, such as elderly care, home nanny, agricultural production, and vehicle driving, trial and error are expensive and a reinforcement learning process that requires much trial and error is difficult to achieve. Therefore, in order to achieve Artificial General Intelligence(AGI) that can be applied to any field, we need to use both existing technologies and solve the defects of existing technologies, so as to further develop the technological wave of artificial intelligence. In this paper, we analyze the limitations of the technical route of large models, and by addressing these limitations, we propose solutions, thus solving the inherent defects of large models. In this paper, we will reveal how to achieve true AGI step by step.

Keywords artificial general intelligence · AGI · AIGC · reinforcement learning · large model · ChatGPT

1 Introduction

The current mainstream artificial intelligence model has brought the spark of artificial general intelligence[1], But it's not really universal AI yet. Where is the ceiling of the current ai model? Can "attention mechanism + deep learning" + "reinforcement learning" achieve truly "Artificial General Intelligence"(AGI)? We believe that the current large AI model cannot solve the following serious flaws:

1.1 It can not solve the problem independently.

Artificial intelligence, for example, does not offer to help when it sees its owner fall[7]. This is because machines do not have their own needs, so it is impossible to produce their own goals. Since the machine does not have its own goals, it is impossible to proactively create a task. In other words, large models do not create new processes!

The big model is essentially a programming platform. The programming language used is the natural language[14]. So, no matter how many high-level functions we add to the big model, and no matter how many tools we integrate into the big model, the big model will not spontaneously create new processes. All of its processes are preset, either from program presets or from data statistics. Both approaches are, essentially, "using out-of-set processes for all problems."[8][9][10][11][12], no matter how many if ...else... are in this process. Considering how many possibilities, it is preset and exists in advance. It's not for specific tasks created by the machine itself! Therefore, the machine that makes decisions according to the predetermined process is the "bookworm" machine intelligence. Decision-making is

not flexible and difficult to face the endless unexpected situations in the actual life, which is also the current dilemma of artificial intelligence.

1.2 knowledge cannot be updated in real time.

At present, artificial intelligence uses big data training, and the knowledge cannot be updated in real time. The real-time update of knowledge is crucial for machines that interact with the environment. Because the interaction between the machine and the environment is the process of the machine acquiring new knowledge. If the knowledge acquired by the machine cannot be updated in real time, a machine, facing the same input information, will constantly make the same mistakes[3].

1.3 It cannot be applied to areas that require interaction with the real environment.

In areas that need to interact with the real environment, such as autonomous driving, doing housework, and caring for patients, machines need to build knowledge of interactive decision-making between their behavior and the external environment. And these areas are difficult to make a lot of trial and error, so machines cannot use reinforcement learning and interact to build decision knowledge in these areas in a real environment[2][4]. We hope that in the future, every family will have a machine nanny; all vehicles will be autonomous; robots will undertake all industry, agriculture, services, and the human main job is to enjoy the beauty of life. But the current artificial intelligence technology solution, still can not achieve the above scenario.

2 How to create knowledge?

2.1 How to describe the information contained in a matrix?

Although a matrix may contain a lot of information, we can express all the information in the matrix by establishing a set of coordinate base clusters. If this set of coordinate base clusters is complete and orthogonal, all the information in the matrix. If the coordinate matrix clusters we establish are not orthogonal but complete, then we can also use this set of coordinate matrix clusters to express any information in the matrix. If the coordinate base cluster is not complete, then there are some vectors in the matrix that cannot be expressed through this set of coordinate base clusters, and then we need to increase the dimension of the coordinate base cluster.

If the substrate coordinate clusters are eigenorthogonal substrate, then we achieve the full information of this vector with the most concise coefficients. If the base coordinate cluster is not completely orthogonal, then we want it to be as close to the orthogonal base cluster as the coefficient matrix we obtain at this time is sparse (highly expression). But if we only care about some of the common information in the matrix, we can use the common information mode as the coordinate base. This base is not an efficient expression for the overall information, but for the common information, it is an efficient way of expression (the coefficient matrix is sparse). So, if we live in an information matrix space, when we need to identify, analyze, and generate a wide variety of information, the most important thing is to find a set of substrate coordinates in the information matrix space.

2.2 How do humans create knowledge?

The information that humans can recognize is only a tiny fraction of the information in our world. This is because we humans have a limited resolution of the information. The relative spatiotemporal relationship between the arrangement of A and B atoms on A grass is also A kind of information, but we will not identify it.

So in the process of evolution, humans have developed the Tokens recognition ability. Tokens Is the smallest information unit commonly used by humans, such as a straight line. Tokens In itself is a kind of "world model", it is the smallest "world model" used by human beings to build a magnificent palace of knowledge. In the process of evolution, humans have formed the "pattern recognition" ability to adopt "models" such as Tokens to identify the surrounding information, which greatly improves the energy efficiency ratio of information recognition. It's a gift from evolution.

Therefore, we take the minimum information units used to human beings, such as points, lines, surfaces, colors, texture, curvature, syllables, tones, symbols, touch, temperature, direction, etc. as Tokens, so we humans live in a 4-dimensional matrix composed of Tokens (three-dimensional space + time dimension). For humans, this 4-dimensional Tokens matrix, from the Big Bang to today, contains the whole knowledge.

Humans slowly use a certain symbol (language symbol) to represent the common Tokens combination, which is the concept. Humans use concepts to describe any information in a matrix (vector): chatting, writing articles. These concepts are a set of coordinate base clusters in our information space matrix.

Under such substrate clusters, the coefficient expression of common information (vectors) is sparse. For example, "investor" represents " human family, rich, want to make more money, find someone to help him earn, take risks, sign an agreement, share the income...".

Concepts contain common Tokens combinations, also contain, language symbols. And because linguistic symbols, appearing more frequently, are more representative, they may become the most commonly used entrance to a concept.

Obviously, the human concept is not orthogonal. Humans are accustomed to take the frequent combination of Tokens as a concept. For the Tokens combinations included in the concepts, there may be non-overlapping, partial overlapping and complete inclusion. Those common Tokens that exist in a large number of things, are highly represented but have low resolution and less number, and they represent abstract concepts. On the basis of abstract concepts, more Tokens are added to form more concrete concepts, which represent a reduced range and a higher resolution.

Although such coordinate base clusters, expressing all information, is not efficient. But they can express common information efficiently. For concepts such as "cat" and "dog", there may be a large number of common Tokens that are non-orthogonal. But it is highly efficient in expressing daily information to humans.

And, this is crucial for the generalization of the information. Because the properties of things are essentially a combination of the Tokens properties that make them up. For example, "cat" is a common arrangement of Tokens in space and time, which may include the language, text, sound, images, action, touch and other multimodal matrix information elements. In this arrangement, some of the matrix elements may have a higher weight, because they are more common, and they may all belong to the concept of "animal"."Animal" contains fewer elements and is more applicable, so between "cat" and "dog", their shared Tokens (such as Tokens related to the concept of "animal") can be directly reused. This is the process of information generalization, and it is also the origin of intelligence.

2.3 What is how deep learning works?

In deep learning, the coefficient of each layer of neural network is behind a set of implicit coordinate bases. The neural network from layer A to layer A + 1 is essentially A substrate transformation process from layer A (expressing information together with the base cluster corresponding to the underlying coordinate base corresponding to layer A) to the coefficient matrix of layer B (expressing information together with the underlying coordinate base cluster corresponding to layer B). Then the information is partially compressed or discarded by the nonlinear function. The essence of deep learning is to use the "trial and error method" to find a suitable set of coordinate base clusters, which can dilute the "useful information" coefficient matrix in the input information.

The purpose of the residual network is to reduce the amount of information loss in each layer of the neural network, so that the machine can make multiple transformations, thus having a greater probability of finding the preferred base cluster. The purpose of regularization is to make the hidden base of the intermediate layer neural network as close as possible to the orthogonal system, so as to avoid the influence of dimensions and avoid the emergence of local optimum. It achieves this purpose by forcing the coefficient matrix of the middle layer close to the thinning matrix.

The high-dimensional features created by deep learning are a set of coordinate base clusters in its information matrix. But it does not have the constraint of "using common Tokens combinations". It uses the "trial and error method" under the error constraint, the established coordinate base clusters, more inclined efficient orthogonal expression. It is more efficient at expressing overall useful information, but different from human habits (humans only need to express those common information efficiently), so "deep learning" and "human" communication are "chicken and duck", less than one piece.

In a large model, the attention mechanism, in essence, is to predict the common degree (occurrence probability) of the local statistical knowledge obtained through pre-training, and to become the preferred coordinate base cluster to identify, analyze and generate various Tokens combinations.

Such a base cluster, it is more in line with human habits. So you can communicate in language between big models and humans. Such an overall expression efficiency is not necessarily high, but is more efficient for expressing common information.

2.4 What is the nature of the attention mechanism?

The core of the attention mechanism is a kind of Bayesian inference (conditional probability). The overview can be "the probability of N Tokens combinations, and the probability of M Tokens combinations".

In human language, the combinations of N Tokens are almost endless, and the M of Tokens are endless. Therefore, it is impossible for the machine to solve the problem of "the probability of known N Tokens combinations, and the

probability of M Tokens combinations". In multimode, the problem is even more prominent. Therefore, the machine can only speculate on the probability of the M Tokens combinations after the N Tokens combinations based on a limited number of statistics. This is the nature of the attention mechanism. The weight matrix obtained by pre-training is a finite amount of statistical knowledge. The attention mechanism is based on a limited number of statistical knowledge to introduce the current Tokens with the probability of M Tokens. If in N + M Tokens, some Tokens have a high weight, they often appear together, so they are more likely to be common Tokens combinations.

This is the central mechanism of the attention mechanism, which is a way to find common arrangements of Tokens. Its essence can be regarded as a Bayesian inference combined with neural networks.

Therefore, the deep learning supported by the attention mechanism and the coordinate base clusters created are more in line with the human habit of creating concepts. That's why big models and humans can communicate in language[15].

In the language model, the "common Tokens combination" is the "common word". It contains the organization of common Tokens, which is similar to grammar; it also contains specific "common phrases", and the "common phrases (including grammar)" is much larger than the human ones.

Attention mechanism, very similar to human learning. When we learn the information in a book, "read thin first, read thick" is the same way."Read thin first" is to summarize the frame information, which is an information compression process, and then "read thick again", is on the basis of the frame information, add different details (and other vectors combined into new vectors) to form new knowledge, which is an information generation process[17][18][19].

2.5 What does the big model work on?

In a large model, when the information is input, the inference process of the attention mechanism is the projection process of putting the input vector to the coordinate base cluster. The weight obtained by the attention mechanism is the coordinate value[15].

In the large model, the input Tokens projects the vector into the weight matrix in the first layer, which is a vector decomposition process. Then, the second layer of projection, which is the input Tokens combination and weighting, after the Tokens combination of the pre-trained weight matrix, the projection process (combination to the combination projection). After the operation of multi-layer attention mechanism, the projection decomposition process of multiple input Tokens combination to pre-trained Tokens combination is formed.

The weight coefficient matrix of the output of the last layer of attention mechanism, and the coordinate base cluster implied behind it (with the common Tokens combination as the coordinate base cluster), together form the re-description of the input information (self-attention mechanism).

Therefore, the working principle of the large model is as follows: (1) it takes the Tokens combination of the pre-trained weight matrix as the base cluster, and the weight matrix is the local statistical information obtained from the training material through the trial and error method; (2) it adopts the attention mechanism to realize the projection process of the input Tokens combination to the weight Tokens combination (vector decomposition), and the weight obtained by the reasoning process is the coordinate value.(3) With the vector component, you can find a large number of adjacent vectors, and the next vector corresponding to these adjacent vectors is the output vector. The proximity relation of the vector is shown in the form of the probability of the output vector.

So, the big model is an autoregressive prediction model. However, it performs the coordinate base cluster transformation process on the original input base (each Tokens is a dimension). Convert the original base cluster such as "every Tokens is a dimension" into a coordinate base cluster such as "after the common Tokens combination, as a dimension". Then, the autoregressive prediction was performed.

2.6 Why are big models able to emerge? When did it come up?

Why do big models "emerge"? A very simple truth, for example, when an American comes to China, he can complete the correct translation process through a large amount of common background information (such as personal needs, social structure, etc.), and a medium number of comparison between Chinese and English.

But the big model is like an alien, and there is no common background information between it and humans, and it sees only the way that human information is connected. So it needs to extract the way that human information is connected to predict the development of information. At first, when the sample is not enough, the "information framework" is very different from the human "information framework", so it keeps making mistakes, groping in the dark, and always running. With the increasing number of samples, its "information framework" and the human "information framework" have a higher probability of alignment. But this is not a linear process. For example, before it reaches a

certain threshold, it decrybles the ancient language, groping in the dark, with little progress. At a certain node, if the accuracy rate reaches the threshold, the whole decryption process will be greatly accelerated and quickly completed. This is the "emergence" phenomenon. It is not about intelligence that the machine "emerges", but about finding the right "common way to combine Tokens". Because the criterion for evaluating the ability of a machine is the human standard, its ability emerges when its base is close to the human one.

2.7 Can RLHF ultimately solve the problems faced by large models?

There are two serious problems with the large model:

2.7.1 Illusion problem[20]

. At present, the core capability of large models is to transform the input information to the coordinate base cluster composed of common Tokens combination (vector projection decomposition), which is a base transformation process of information space.

It then uses the obtained coefficient matrix (the inference weight of the attention mechanism) to find multiple similar "pre-trained vectors" (component-weighted contrast). Then, according to these similar "pre-training vectors", follow the mapping relationship obtained by the pre-training, find the "next vector", and select one of the outputs. This is the autoregressive prediction process, and how the large model of the GPT class works. So, the big model optimizes the "parameters". For each parameter, it corresponds to a set of Tokens combinations. On the surface, the large model works in optimizing the network parameters. Its essence is to optimize the common Tokens combinations, that is to say, looking for a set of optimal base coordinate clusters. Each layer of coefficients of a neural network corresponds to a cluster of underlying substrate coordinates.

Large models have only "common Tokens combinations" derived from huge amounts of data, and have no factual memory. Therefore, facing the input Tokens, the large model can only decompose the input information to the "coordinate base cluster", and then obtain the next Tokens with different probabilities. This process proceeds iteratively, and it in itself is a creative process. If the fact itself is "common," then the fact is retained in the form of "common Tokens combinations". If the facts are not retained as a "common Tokens combination", or the facts themselves are not weighted enough, then the machine creates information. The GPT itself is information generation, so the hallucination problem is a part of its job[17][18][19], So the GPT has no solution to this problem.

For example, the machine finds that behind the profile of many journalists, there are links to other articles, or awards that they have won in the past. If the machine sees this pattern of information organization, then this mode of information organization becomes a mapping from "framework" to "framework". So if the input information contains a similar frame, but only the reporter's name is different, then the machine can map to the "frame + details" through the "frame + details", which can also produce a lot of web links, or awards in the output. But these web links and awards are also built by mapping "frame + other vectors" to "frame + other vectors", and they probably do not exist at all!

In order to solve the illusion problem of the big model, many people expect to plug in the "vector database", let the big model to query the factual knowledge to eliminate the illusion. This is another version of an attempt to adopt the encyclopedia to implement general AI. Whether it is a "vector database" or a "knowledge graph", it is impossible to solve the illusion problem! Because, these knowledge is plug-in, and the knowledge of the big model itself can not be integrated. They are like an ordinary person taking a dictionary and trying to open a translation company. When the expert system encountered the problems, it will encounter.

2.7.2 The question of the harmful content[20]

In large models, the attention mechanism is correct, but deep learning is flawed.

In the large model, the Transform model based on Selfattention, including position coding, its main purpose is to increase the Tokens position information, so that it can use the position relationship between each other. This is necessary for the attention mechanism, because it is to find the temporal and spatial relationship of the Tokens.

However, through the multi-layer deep learning network, the large model finds the "optimal coordinate substrate cluster" after multiple coordinate substrate transformations under the error constraints. However, this Tokens combination of the "optimal coordinate base cluster" is no longer the same as the temporal and spatial relationship of the original Tokens. While it may still retain some of the organizational information between Tokens (because the deep learning process is irreversible, the location information of Tokens is only partially retained), it is difficult to be understood and exploited by humans. So, we believe that deep learning destroys the original temporal / spatial organization form of Tokens.

We can think that the large model performs a lossy translation process, translating the human Tokens into its language. However, the problem is that human beings do not master the language of the large model, so human beings cannot understand the knowledge created by the large model, nor can they imitate the form of knowledge organization, and implant "innate knowledge" into the large model. This is the core of the problem.

Moreover, because the large model cannot realize small samples and cumulative learning, it needs very large samples and takes the knowledge shape at once, which further increases the difficulty for human beings to understand the form of knowledge organization. Because machines have no own needs, they cannot have self-perceived rewards and punishments. Without self-perceived rewards and punishment, it is impossible to spontaneously create a projection of the vector (information) to the reward or punishment dimension. That is to say, in the base coordinate cluster created by the machine, the lack of rewards, punishment, happiness, sadness and other human unique, also must have the basic dimension!

The current remedy used by the large model is the RLHF. This is equivalent to humans adding a suffix of a reward dimension to a particular vector. That is, in the base coordinate cluster of the machine, a reward dimension is added. If in the training data, increasing the component value in the reward dimension on a large number of different types, sufficient number of vectors, it is equivalent to establishing the common component combinations in these training vectors, projections to the reward dimension. This is the reward function of the machine. So, the machine can also predict the reward component contained in the output vector produced in different decisions, namely in different combinations. Therefore, the machine will prefer the output with a high reward component. This is the amazing effect of RLHF learning. Because the knowledge learned through the RLHF can actually be generalized. When a machine has its own dimension of reward and punishment, it has its own preliminary "consciousness of" seeking benefits and avoiding harm ", which is why we can see the hazy shadow of" consciousness " from the current big model.

But it's a patch, which means the machine to try, then humans score and feedback, and it can only be used in areas where there is a lot of trial and error. This is similar to a child who graduated with a PhD, but there is no concept of "right and wrong", the parents can only shout "No", "No", "Yes" to give him the concept of "right and wrong" behind, and he can't communicate directly with his parents, only through "Yes" and "No" to communicate. Therefore, this learning effect is low efficiency, and may always encounter those unexpected corner case!

3 Is attention mechanism + deep learning + reinforcement learning the right path for artificial general intelligence?

3.1 Can the big model achieve AGI?

We believe that the large model proves its general direction. But we don't think that large models are the right way to achieve general AI.

In terms of NLP, humans range from early bag model, word vector to EMLO[21], Until Transformer, the attention mechanism is truly realized. After combining deep learning and attention mechanisms[22], Can produce optimized coordinate base clusters similar to human expression, which is why Transformer can produce intellectual "emergence".

However, we note that the path of the large model is "to establish the preliminary relationship; then adjust the coordinate base cluster; then under the preferred coordinate base cluster to obtain the correct relationship". Such a mechanism leads to a huge amount of data required and computation, and knowledge is formed through the training process, which is difficult to update in real time[2][3].

At the same time, the reward function appears after the event, which is not applicable to difficult trial and error areas, such as interactive decisions in real environments (autonomous driving, home nannies, industry, agriculture, business, services, government management, etc.).

In addition, the idea of "task-oriented, do reinforcement learning" is wrong. The reason why human beings are "universal" is that we face all tasks and make decisions according to "seeking advantages and avoiding disadvantages". The ines should be the same. There are thousands of tasks, task-oriented reinforcement learning, never learn! And many tasks cost very much for trial and error!

3.2 What kind of road is the right way to reach the AGI?

The current problems with the big model are described in this way:

(1) The attention mechanism is right. But deep learning is flawed.

Because deep learning destroys the original form of temporal / spatial organization of Tokens. The knowledge generated is difficult to understand and cannot be imitated. So humans cannot imitate their organizational form and implant innate "self-needs" (innate knowledge) into machines.

Without "self-needs", it is impossible to have "own ideas" and "independent decisions". In this way, the machine can only follow the predetermined process (or preset, or statistics), passive "decision", not flexible, which is the big problem of AI at present.

(2) The idea of "task-oriented, do reinforcement learning" is wrong.

The reason why human beings are "universal" is that we face all tasks and make decisions according to "seeking advantages and avoiding disadvantages". The ines should be the same. There are thousands of tasks, task-oriented reinforcement learning, never learn! And many tasks cost very much for trial and error! Like taking care of children, no one wants to give their children to a machine for experiments!

So, our solution is to:

(1) Realize the attention mechanism without destroying the original temporal / spatial organization form of Tokens. The knowledge created can be understood and can be imitated.

(2) We can imitate the organizational form of knowledge and give "with" innate needs "."Innate needs", as a special class of Tokens, and other Tokens, form common combinations through attentional mechanisms. These common combinations are common sense (that's the world model)!

(3) The machine only learns one thing "how to meet its own needs", and only deals with one thing "how to meet its own needs". This is the general decision-making.

(4) Because the original temporal / spatial organization form of Tokens is not destroyed, the machine can directly obtain the temporal and spatial arrangement of Tokens through language symbols. And this arrangement can be understood and imitated, so that machines can directly acquire all the experience accumulated in the history of human civilization through language learning! Machines no longer need to go through the "evolutionary history"!

4 Step by Step steps for implementing general AI.

Here are the 10 steps to implement our protocol.

Step 1, Tokenize information.(Like any other AI technology)

Step 2, Matrices the Tokens.(Build a memory data library)

Step 3, The input Tokens propagates the activation value to the Tokens in the memory bank according to the similarity relationship.

Step 4, All the activated Tokens, following the proximity relationship, propagate the activation values to the adjacent Tokens.

Step 5, Each activated Tokens, in turn, spreads the activation value in the memory bank.

Among them, from Step 3 to Step 5, the higher the similarity, the greater the transfer coefficient. The closer the storage location is, the larger the transfer coefficient is. The higher memory value of Tokens indicates a larger transfer coefficient.

Step 6, The activation value obtained for each Token from different propagation paths, is accumulated.

Step 7, The activation values of all Tokens, were resolved over time. Among them, Step3 Step7 is the process of chain association activation, which is the inference process of the attention mechanism, and the activation value is the inference weight.

Step 8, Each Token updates the memory value according to the size of the activation value obtained. And, all memory values fade over time. Each Token's memory value is its pre-trained weight value. In memory, there are a large number of Tokens combinations, those that can appear repeatedly, which contain Tokens that can activate each other each time to push up the activation value, thus obtaining higher memory values. So if a combination of multiple Tokens's appears in the input, the Tokens combination has a higher probability of getting a high attention weight. Therefore, the chain of associative activation process is a "Tokens combination" first activation value propagation process.

Step 9, preset minimum innate requirements (innate knowledge, composed of Tokens + memory value + arrangement). Innate demand are the organizational form of imitation knowledge and the establishment of innate knowledge. Innate

Table 1: composition of each Token data

Filed1	Filed2	Filed3	Filed4
Record Time	Token	Memory value	Activation value

knowledge can include the minimum innate needs, rewards and punishments, emotions, and the necessary innate safety instinct knowledge, and of course, other knowledge can also be preset. This knowledge exists as part of the memory bank and seamlessly integrates with the acquired memory to form the overall memory bank. The "Fine Tuning" of innate knowledge is achieved by the accumulation of acquired knowledge (including feedback).

Step10, Let the innate needs, rewards and punishments and emotions (using special Tokens to represent), and acquired information (ordinary Tokens information flow), in the machine training and life, the formation of time information flow, and is stored. Then, through the chain association activation process + attention mechanism, a fully connected knowledge network (memory bank) is formed. Our scheme ends up with a memory bank: each Token is a data record. They consisted of the 4 fields shown in Table 1. Time mark: Represents the temporal relationship of the Tokens to each other

Token: Represents the Tokens itself, can be data from graphics, voice, or other sensors.

Memory value: Represents the pre-training weights.

Activation value: Reasential weights representing the attention mechanism. A large number of Tokens are stored at time intervals, and a knowledge network is formed through optimization (through the chain association activation process + memory and forgetting mechanism to survive the fittest).

Knowledge network, it is the memory bank. The network node, which is the Tokens. The network connection, which is the activation value transfer relationship. However, it should be pointed out that the activation value transmission relationship is determined by the relative position of Tokens, the similarity between the memory value of Tokens and Tokens, and the size of the initial activation value obtained by Tokens. Therefore, the Tokens is input first, and then the activation value transmission relationship between Tokens is temporarily established. This transmission relationship is not fixed.

The memory value represents the pre-training weight; the activation value represents the reasoning weight under the attention mechanism. So, in our scheme, knowledge acquisition and reasoning application are integrated, and innate knowledge and acquired knowledge are integrated.

In the memory bank, there are both objective Tokens and subjective Tokens, and the connections formed through the attention mechanism is "information". The permutation of all Tokens is all information, with has high dimension. And "knowledge" is the arrangement that can repeat (including time, space), they are the part of the information that can be repeated, so they contain less Tokens, more representative, more applicable, more abstract, so they have less dimensions. Common sense is further limited to our common human "knowledge".

Our machines, memory banks can be inserted, modified, or merged, so knowledge between machines can be shared directly by memory banks. For example, a chef robot, by loading the doctor robot's memory, can directly acquire the doctor's various skills. There is no need to combine the "chef big data" and the "Doctor big data" again, and spend tens of millions of dollars and several months to redo the pre-training.

4.1 Detailed description of each step

4.1.1 Step1, Tokens Tokens the information

The machine only needs to disperse the input information, according to the overall priority, according to the low resolution priority, extract the underlying Tokens (such as the overall outline, texture, topology, line, image, horn, ridge, vertex, voice time domain / frequency range tone, timbre and other main underlying Tokens).

In chronological order, then stored in the memory bank is OK. Special emphasis: no need to identify them, save it OK. Even if the Tokens extracted is random, the algorithm is imperfect. Because our algorithm is based on accumulating common Tokens combinations (the "world model"), which guides the machine on how to "extract on demand"! Common Tokens combinations contain both common Tokens and their organizational forms.

So the process of Tokens extraction is a step by step optimization process. After the Tokens is stored in the memory bank, the memory value and activation value of these Tokens are constantly changed according to the chain associative activation process, memory and forgetting mechanism. Through survival of the fittest, those widespread Tokens, or

Tokens combinations, will be retained to form a more complex Tokens. Those Tokens that rarely repeat are eliminated and they are no longer extracted.

So, the strategy of machines to process Tokens is also: look for the widespread combination of raw data, as Tokens. This is the application of the common information combination first principle in determining the Tokens composition. This, similar to humans, is a gift from evolution to humans. Because extracting underlying programs such as Tokens requires extensive multiplexing to achieve maximum energy utility.

4.1.2 Step2, which matrices the Tokens

Each Token, corresponding to a record in the memory bank, has 4 fields, as shown in Table 1. Memory value size indicates memory strength, and zero is removed. Activation value size indicates the strength of being activated, and being zero indicates not being activated. All the records spontaneously constitute the entire memory bank according to the simultaneous storage method. As for the simultaneous storage method, the specific embodiments include: (2.1) The machine retains the time relative position where the Tokens appears in the input information.

One implementation method is: the machine uses the distance of the Tokens in the storage space to reflect the time distance between the time when the Tokens are stored, for example, the machine stores Tokens in order according to the input time order, the closer the time Tokens, the closer the closer the storage position;

Another method for storing the relative position of the retention time is that each Tokens has the coordinates in the memory space, mainly including the storage time information of the Tokens;

The machine retains the spatial relative position of Tokens in the input information; one implementation method puts the extracted Tokens in the overlap with the original data, and keeps the relative position of the Tokens in space during storage;

The implementation method can also be: the overall low-resolution Tokens priority extraction, and then based on the machine's decision, and then extract other local Tokens on demand. In this way, through the proximity storage relationship, local Tokens and overall Tokens have both proximity activation relationship and similarity relationship between Tokens, so they will activate each other and establish positional connections.

4.1.3 Step3, from the input Tokens to the Tokens in the memory bank, performs similarity activation

Each Token of the input is given a uniform initial activation value of A0. The A0 in itself is a preset numerical value. However, it can be adjusted by the activation value of the activated reward symbol and the punishment symbol during the last machine chain association activation process.

The activation value of the activated reward symbol and the punishment symbol is the potential reward and penalty value predicted by the machine for the previous input information. The initial activation value, A0, affects the range of the chain association activation process. When the initial activation value A0 is high, the spread of the chain association activation process is larger. This is because in our scheme, the activation value propagation coefficient is less than 1. As the number of chain spread increases, the value of spread becomes smaller and smaller. The chain propagation process ends when a Token obtains an activation value less than a preset threshold. So A0 reflects how much the machine values the input information. When A0 is high, the machine activates more Tokens in memory to look for Tokens. This is similar to humans, where if the previously input Tokens brings high potential rewards and penalties, then the new relevant Tokens inputs may be particularly valued. For example, the boss, will let you associate more information.

The principle of similarity activation is: (1) the higher the similarity between Tokens, the greater the transfer coefficient; this is the point product of the correlation between Token.(2) The higher the memory value, the greater the transmission coefficient; the memory value is the pre-training weight. It should be emphasized that the same Token may constantly appear in many locations in the memory bank! They all have their own different memory values! This is because the same Token in the same Token weight is not the same! This is similar to the attention mechanism in the large model.

4.1.4 Step4, all activated Tokens, performs proximity activation

We believe that the proximity relationship between Tokens represents some implicit correlation between them. The closer the occurrence time is, the closer the potential relationship is. This correlation can be counted by the chain of association activation process + memory and forgetting mechanism. The proximity relationship actually reflects a Tokens combination relationship. If this combination relationship repeats itself, then it is a common combination. So we find out the common combination through the near activation process during the chain association activation process.

Each activated Token then transmits the activation value to its adjacent Token; the closer the time position, the greater the transmission coefficient; the higher the memory value, the greater the transmission coefficient. If there is a close relationship between Tokens in memory banks, it shows that they were once a combination way. If their memory values are high, it indicates that they are a common way of combination. If only one Token has a high memory value, they are not common combinations. If neither memory value of the Tokens is high, then their propagation activation value is low and the chain propagation of Tokens stops quickly. It means that such information is not important and they have a low weight in information processing. Token adopts the method of "the closer the time position, the greater the transmission coefficient; the higher the memory value, the greater the transmission coefficient" to activate the common combination containing them, which is essentially the projection process of inputting Token to the coordinate base composed of a set of Tokens.

If the N input Tokens are projected onto the X combination (Tokens combination) in memory, then the X combinations get a high activation value. Because each Tokens will activate multiple Tokens in the X combination with both similarity and proximity. Therefore, the X combination achieves a higher activation value by accumulating the activation value. These higher activation values of Tokens, composed of the "model", is the expected model of the input vector (N Tokens) activation (world model).

In essence, this is a process of vector decomposition to coordinate substrate, and also an information recognition process.

4.1.5 Step5, each activated Tokens, in turn spreads the activation value in the memory bank following the similar and proximity activation principles

Each input Token has "similarity activation" and "proximity activation" in the memory bank, and the size of activation value transmission is positively correlated with their pre-training weights (memory values).

Each activated Token in the memory bank is also positively correlated with "similarity activation" and "proximity activation", and the activation value transfer size is positively correlated with their pre-training weight (memory value).

This process proceeds strand until all the inputs Token complete their own "chain activation process". So, in addition to the activation of the memory vector similar to the input vector, the machine also activates the "antecedents" and "consequences" of the memory vector similar to the input vector, that is, the front and back information in time in the memory bank. And, possibly through different memory segments, activate different "antecedents" and "consequences". This allows our scheme to speculate on the possible former vector and to predict the possible next vector.

Since the strategy we adopted is "the overall low score rate Tokens" first, the spatial position relationship of the information is actually established through the temporal position relationship. When the information is entered, the machine first extracts the "overall low score rate Tokens", which stores it in the memory bank. A chained associative activation process is subsequently initiated. After completion, the decision is made by counting the activation value of the activated reward and penalty symbols.

The decision-making principle of machines is to seek advantages and avoid disadvantages. Decisions may be made by further identifying information, or other decisions. If the decision is to further identify the information, then the machine will take the current high activation value Tokens combination mode (including the language Tokens) as the expected model to proactively confirm the high activation value Tokens that has not yet appeared in the input. The method used is to imitate the past experience of obtaining these Tokens to adjust your own sensor system. So this is a "pattern recognition" for information, which is similar to the human recognition process.

These newly acquired Tokens (such as local details) have a temporal proximity relationship with the original "overall low score rate Tokens", but also have a partial similarity relationship, so they can be connected by passing activation values to each other. In this way, the newly acquired Tokens establishes a positional relationship with the original overall low score rate Tokens. These overall low score rates, Tokens, and those often accompanying local Tokens, slowly form a "world model" through memory and forgetting mechanisms.

It should be pointed out that the world model is not to create an independent model, it contains Tokens that may spread throughout the memory bank, and these Tokens are temporarily created by similarity, proximity, and high memory values. So it is not static, it is distributed existence, it is temporarily composed of those Tokens with high activation values, motivated by the input information, and there is no separate model in the memory bank.

4.1.6 Step6, the activation values are accumulated

If there is an activation value propagation path between a Token and multiple input Tokens (i. e., either directly or indirectly), the activation value passed from the input is cumulative. So Token, a memory bank with multiple input Token, will obtain higher cumulative activation values from multiple propagation paths.

In this way, the input in Token, if the Tokens is associated to each other, it pushes up the weight of the related Tokens in the memory bank. That is, the common combinations, their activation values, rise from the activation values to sea level. And this activation value at sea level is the low activation value of those with large amounts of Tokens. Those Tokens that rise from the activated sea level constitute one or more "world models". While those memories most related to the input, although they may not be common, may also obtain high activation values because they are directly related to the input and have short propagation paths.

Therefore, our scheme can not only obtain the "information framework" of information through the common Tokens combination, but also pay attention to specific factual details, so our scheme, is a "fact database", which can solve the current "illusion" problem of GPT.

4.1.7 Step7, activation values subsided over time

All of the activation values are constantly decreasing over time. When the following Token input, the memory related Token is activated. In the previous input, the activated Token has not completely subsided, the activation value will be accumulated.

The machine’s decision is based on all the activated Tokens. So both the first and back inputs are taken into account. Therefore, the thinking of the machine has a certain time consistency, which can solve the problems of "omission", "reference" and "metaphor".

So, our machine takes advantage of the implicit relationship between the front and back inputs! That’s the attention mechanism!

Further: the machine adjusts the initial activation value A0 assigned to the input Token s based on the "pros and cons" predicted by the last decision. The initial activation value A0 will affect the range and cumulative size of the propagation of the activation value! This is to adjust the intensity of attention according to the "pros and cons"! It’s very similar to humans. At this point, beyond the current technology (Transformer). In fact, this is very similar to the human decision-making process, such as the boss, which allows you to generate more associations, activate more reward or punishment symbols, so as to predict rewards and penalties more deeply.

4.1.8 Step8, to update the pre-training weight matrix through the chain association activation process + memory and forgetting mechanism + the principle of seeking benefits and avoiding disadvantages

In our protocol, those Tokens combinations that are able to recur are likely to obtain higher memory values because of repetition. And because it is a recurring combination, each other pushes up the activation value, so it obtains a far higher memory value than simple.

And because they can repeat, their combination achieves a higher activation value each time, so they are easier to be activated, and thus easier to gain memory increments. So this is a positive cycle process. So, from here we can see, our machines can learn from themselves. But at the same time, forgetting the pre-existing mindset is also a time-consuming process.

So, in our protocol, the pre-training statistical process of the machine is not simply statistical repeatability and then using memory and forgetting mechanisms. But through the attention mechanism + memory and forgetting mechanism + benefit and avoid the principle of the joint completion.

The decision-making process of the machine is to seek advantages and avoid disadvantages. In the decision-making of seeking advantages and avoiding disadvantages, the process of identifying information according to the way of seeking advantages and avoiding disadvantages. So, our machines, based on their own needs, to build common combinations of Tokens. So, our machines, to the outside world, to their own information recognition, are selective recognition.

Memory and forgetting mechanism: all Tokens in the memory bank, if activated once, will positively update their memory value according to the size of their activation value. Their memory value is the pre-trained weight matrix! Since the Token permutation cannot be exhaustive, this is a non-complete statistical process, which is similar to the pre-training process of large models.

The chain association activation process is the inference process of the attention mechanism (from the local statistical weight to the input localization weight calculation process) under the input Token combination incentive, which is similar to the attention mechanism in Transformer.

This process is essentially the projection of the input vector to the coordinate base cluster established by the attention mechanism. The input vector, which can be regarded as the original base cluster formed by the pulse function under the input dimension. However, the coordinate base cluster established by the attention mechanism is established on the basis of common Tokens combinations.

The inference weight matrix of the attention mechanism is the coefficient matrix of the projection of the input vector to the base cluster. In our scheme, the chain association activation process, similar to the multi-layer attention mechanism in Transformer, is also the process of the coordinate base cluster projection to the attention mechanism: first separate Tokens projection, and then combined projection. After the final chain activation, one or more high weight components of the high activation value Tokens is the "framework" of information. Each framework contains many Tokens, which is difficult to describe specifically. But usually the language symbols, due to high representativeness and repeatability, so the activation value may also be the highest, so they may become the representative Tokens of this "framework". So, in our scheme, the activation value is the inference weight matrix.

In fact, both the large model and our network are a kind of neural networks. Attention mechanism, in essence, is Bayesian inference. Generally speaking, the attention mechanism is the conditional probability of some Tokens and the joint probability of some Tokens, and the joint conditional probability of a specific combination of Tokens. This is the application of combining Bayesian inference and neural networks. In the large model, it is known that the probability of some Tokens, and the joint probability of some Tokens are determined by the weight matrix, and the probability prediction under Tokens combination is performed through multiple correlation operations. In our scheme, it is known that the joint probabilities of some Tokens and some Tokens are explicitly expressed in the memory bank, which are the memory values of Tokens, the relative position of Tokens and the similarity between Tokens.

As you can see, the way we realize the attention mechanism is a small sample, cumulative learning. And the weight matrix is updated in real time, so our scheme, knowledge is updated in real time. And we don't distinguish between pre-training and reasoning processes, so our machine is lifelong learning. In addition, it can be seen that our scheme does not need BP algorithm, does not need pre-training, and its basic operation amount is close to the reasoning process of the large model. Therefore, the computational amount of our scheme is much less than Transformer, and it can also be calculated in parallel. Therefore, our scheme can realize the computational localization of the pre-training process. Every machine is a self-training, constantly iterative, and constantly evolving agent.

In addition, it can be seen that in our scheme, Tokens extraction can adopt similar techniques to the current large models, and the amount of calculation is comparable. The chain association activation process is highly stereotyped, which can be directly implemented at the hardware level with new memory devices. In this way, it will help in the localization of the calculation in our scheme, which will help to expand the landing scenario and reduce the cost.

4.1.9 Step 9, preset minimum innate requirements

We not only realized the attention mechanism, found the common Tokens combination, but also did not disrupt the original temporal and spatial organization form of Tokens! Therefore, the knowledge network formed by our scheme is understandable by human beings. So, we can imitate the organization form of Tokens in the final memory bank and build the initial minimum congenital memory for the machine! This is equivalent to giving a machine a minimum of innate knowledge similar to humans (what a baby is born with).

In innate memory, the minimum "demand system", "reward and punishment system" and "emotional system" of the machine are needed. The method is: use special Tokens to represent each "demand", "reward and punishment" and "emotion". Then imitate the form of pre-training (in fact, the appropriate Tokens arrangement + the appropriate memory value) and implant the minimum innate knowledge.

In daily life, let these Tokens, which represent "demand", "reward and punishment" and "emotion", and other external Tokens that trigger them train together, activate the chain association together, and remember and forget together. That is, through the attention mechanism, let these special Tokens, like other Tokens, establish common Tokens combinations. Therefore, we must preset the minimum "demand system", "reward and punishment system" and "emotional system" of the machine, so that the Tokens representing the outside world (including the state parameters of the machine itself) can trigger these special Tokens, so as to establish the information flow. And through the chain association activation process + profit and harm avoidance decision + memory and forgetting mechanism, to gradually obtain the most common, and the machine is most concerned about the common Tokens combination.

In this way, we establish a connection between the "common Tokens combination of the objective world" and the "needs". The "common Tokens combination of the objective world" is the "objective common sense" of the objective world, and the "common Tokens combination" composed of the "common Tokens combination of the objective world" and the "demand" is the "subjective common sense"."Objective common sense" and "subjective common sense" constitute "common sense".

Common sense is the "world model", which contains the "world model" of human cognition of the external world, and also contains the relationship between the "world model" and "I" established by human beings. In particular, it should be pointed out that Tokens is not only static features, but also contains those simple dynamic features (such as rotation, swing, etc.), so the world mode is not static or fixed, but is created under the input Tokens excitation!

And each world model is different, which is directly related to its experience. In our scheme, the "world model" built by the machine is directly related to its training data, as well as to its life experience!

With the world model, the input Tokens can activate the reward and punishment Tokens, emotion Tokens and demand Tokens through the chain association activation process, and the transfer path of the activation value from the input Tokens to these features Tokens is a logical reasoning process compatible with the neural network! It is explicit, is understood, can be imitated, so the machine decision can be seen.

In fact, Step 9 is essentially the first step in the actual creation of a general AI. But we can train the experimental data through the previous steps, and thus obtain and understand the organizational forms of the knowledge created by the machine, and then imitate these organizational forms to implement Step 9.

(1) Preset and machine life activities related, basic requirements pros and cons system. For example, give a reasonable interval to the battery data, preset a symbol representing "hungry" in the "innate memory", and put a "punishment" symbol and an emotional symbol representing "hungry" next to the "hungry" symbol. And give them the appropriate memory value.

When the power is insufficient, the vital state monitoring program will directly give the initial activation value to the symbol of "hungry" in the "innate memory". Its activation value will spread chain throughout the memory bank. The "hungry" emotion symbol next to it is activated, and the "punishment" symbol next to it is also activated. So the machine has a "hungry" mood and a "punishment value". In order to avoid the "penalty value", the machine will use its own experience to actively find a plug to charge!

(2) The advantages and disadvantages of the "higher order needs" of preset machine values need to preset the simplest means of communication and then cultivate values.

Values need to be cultivated from childhood! So we need to educate us about the "values" of robots from an early age. Since education, it needs to be achieved through "reward" and "punishment". So when the machine starts out, it needs to be able to recognize "rewards" and "punishments". In this way, we can initiate the first step of learning through "reward" and "punishment"!

Therefore, we need to imitate the acquired memory network organization form, so that the machines can have the innate knowledge that can recognize the simplest "reward" and "punishment"!

For example: preset the most basic head nodding features (assuming X Tokens) / head shaking features (assuming Y features), do not need to be accurate!

Next nodding Tokens, place a "respected" symbol; place a "reward" symbol; give these symbols a higher memory value, and make their relationship a long-term memory. When a partial nod Tokens appears in the information input, the machine obtains the "reward value" through the chain association activation process. In pursuit of "reward", the machine may plan various decisions in the future of obtain "human nod"!

Similar to a child, starting from the simplest way of communication, gradually acquire complex learning ability, he (she) gradually established the "reward function" logical chain is: "milk", "pacifier", "bottle", "milk powder can"...→ ...."Academic performance", "house, car"..."Social status"....→ "worldly ideal".

Therefore, after training, there are a large number of reward and punishment related Tokens symbols in the memory bank of the machine, and Tokens combinations closely related to these reward and punishment Tokens, there is a causal relationship between them. These Tokens combinations that are closely related to reward and punishment Tokens, which represent things, behaviors and results, are values. Therefore, any value of the machine can be established by preset innate communication means, and then cultivating it step by step. In fact, human beings are the same, no one is born with a "saint".

Figure 1 is a schematic diagram of the "innate minimum requirement".

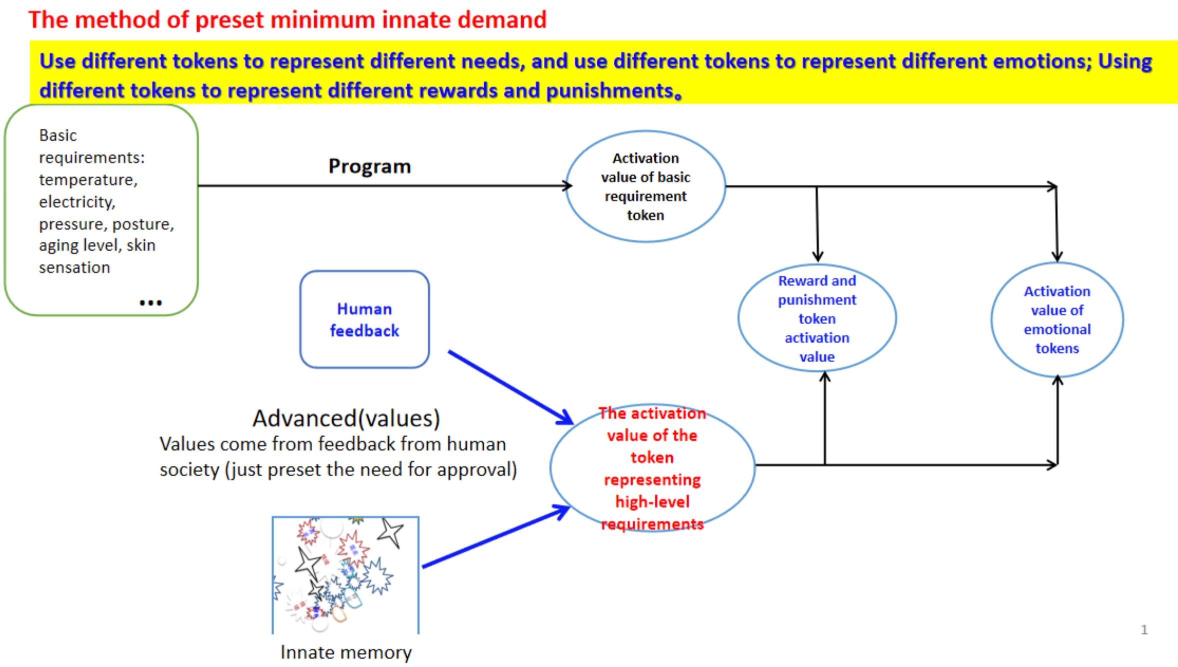

Figure 1: establishes a schematic diagram of the "innate minimum requirement".

4.1.10 Step 10 To form a fully connected knowledge network.

Our scheme ends up with a network where each Token is composed of four fields: the time mark, the Tokens itself, the memory value, and the activation value.

A large number of Tokens are stored according to the time interval, and through the optimization (using: chain association activation process + memory and forgetting mechanism to survive the fittest), the knowledge network is formed, in which the memory value represents the pre-training weight; the activation value represents the reasoning weight under the attention mechanism.

Our networks have both objective Tokens and subjective Tokens, and their connections through the attention mechanism are knowledge, among which common knowledge is "common sense". This is why our machines can predict the pros and cons and make their own decisions! Because it has a "demand", and a "logical chain" related to the "demand" (the activation value transfer link formed by Tokens). Driven by the demand, it will take the initiative to learn and iterate on itself! For example, go to recharge, go to the library to read!

In our scheme, knowledge is developed around "demand", and decisions are also developed around "demand", which is the core reason why our machine can achieve "universal"! It faces only one task: "needs", rather than all kinds of "external tasks". So, our solution is "active wisdom", and all other solutions are "passive" wisdom.

As you can see, our scheme is small sample learning, real-time knowledge update, and the training and use process are integrated, so the machine is lifelong learning, self-iteration. Because the knowledge of the machine exists in the form of a memory bank, and the memory bank is stored in chronological order, but on the basis of the original memory bank, the memory value is gradually optimized. So different memory banks can be directly stitched together to form large memory banks. So, by combining the chef's memory bank and the doctor's memory bank, the robot can have the skills of both a chef and a doctor, without having to retrain a lot of a chef and a doctor's data together. The current AI technology route, can not achieve this point. In large models, it has to be trained with a large amount of both doctor and chef data for a machine to master both skills. Obviously, with this training method, it is a wild hope that machines can have "all kinds of" abilities.

4.2 An example of the process of changes in memory values and activation values

Figure 2 shows a simple example of the process of changes in memory values and activation values during associative activation. To simplify, assuming that the machine's memory bank is empty, it is the first time the machine receives the

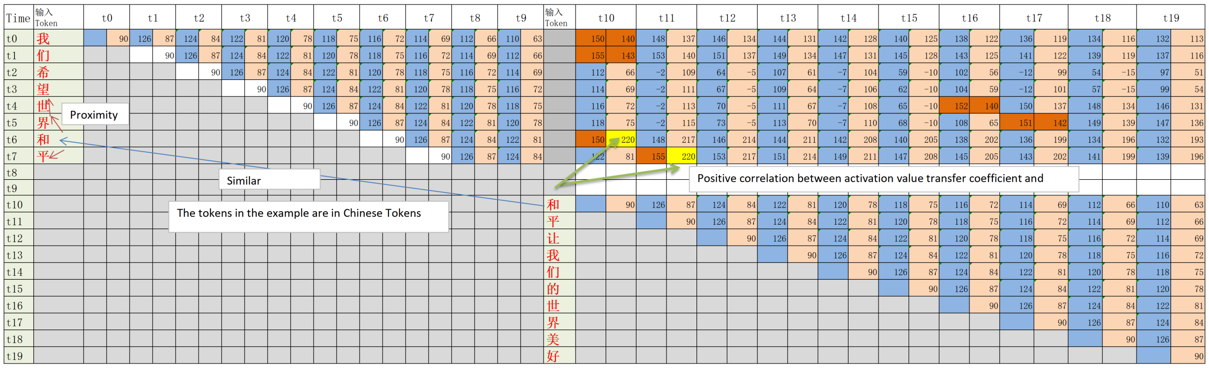

Figure 2: establishes a schematic diagram of the "innate minimum requirement".

input (and we don't give the machine a preset innate memory). Suppose that, at time t0 to t7, the machine input Tokens is "We hope for world peace". In the actual process, the machine should adjust the initial activation value to all input Tokens according to the activation value size of the currently activated reward and penalty symbol (value estimate). But here, because there is no value system to adjust, we assume the default input Tokens initial activation value of 90 (assuming activation value interval is 0 255), so after the chain association activation process, assuming the memory curve, Tokens activation value is 90, and the current memory value of 0, the machine obtained memory value increment is 126.

dm = f(m0, A0) Memory values update increments, where m0 represents the current memory value and A0 represents the current activation value. Memory value update increments and activation values are positively correlated.

All memory values and activation values decrease over time. Here, an exaggerated descending gradient was employed. At time t9 to t19, the machine receives the second input Tokens: "Peace makes our world better". Obviously, according to the "similarity activation" process, first Token "and" will activate Token "and" in the memory bank, and give it activation value, and because the memory value of "and" in the memory bank is high, so the "and" in the memory bank from the initial activation value, obtained the activation value of the past, and the transfer coefficient is large.

The similarity activation process transfer coefficient T=f(S,m0), where S represents the similarity (the dot product of the Tokens vector), and m0 represents the memory value of the transmitted Tokens. Positive correlation between activation value transfer coefficient and similarity and memory value.

At the same time, the "and" Token in the memory bank will also initiate the chain propagation process because the activation value exceeds the preset threshold. In the process of chain propagation, it will first activate the near relationship to the "flat" and "bound" through the "near activation" way.

The near activation process transfer coefficient T=f(D,m0), where D represents the temporal distance of the two Tokens and m 0 represents the memory value of the transmitted Tokens. The near activation value transfer coefficient and time distance are anti-correlated, and the memory value of the transmitted Tokens is positively correlated.

After the "flat" and "bound" obtain the activation value, if the activation value exceeds the preset threshold, the chain will also initiate the propagation process. Looking for the Tokens similar to oneself in the memory library to conduct activation value propagation, it will also conduct activation value propagation to the Tokens adjacent to oneself. The transfer coefficient of both processes is positively correlated with the memory value.

Through the chain association activation process, with input Tokens, it is possible to activate the entire memory repertoire and their associated Tokens combinations. The activation range depends on the initial activation value they obtain, which is adjusted by the value prediction.

After completing the chain association activation process in the second input, we can see that in the Tokens stored in the memory bank, the "peace" Tokens combination has the highest memory value and approaches, so each Token will get a higher activation value due to the high memory value. At the same time, "and", "flat" in addition to oneself have a chance to get higher activation value, they will also pass each other activation value (near activation), and the process, also because of their high memory value and high transfer coefficient, through activation value accumulation, they are a set of easy to obtain high activation value weight of Tokens combination.

Second, we can see that in the Tokens stored in the memory bank, the "world" Tokens combination is similar to the "peace" Tokens combination, obtaining the second highest memory value, so they are also Tokens combinations that are prone to obtain high activation value weights.

In Figure 2, we use Chinese characters to represent a token, and of course, we can also use word vectors to express a token, without any difference in the entire process.

So, we only need two sentences, we can establish the relative "weight" of Tokens. By doing this, the machine can build the right combination of common Tokens and their memory values. This memory value corresponds to the "common degree" of the combination, that is, the memory value is actually the local statistical value of the occurrence probability of the combination obtained from the training data. The activation value is the dot product process (projection) of the input Tokens combination to the "common Tokens combination" (base cluster) based on this local statistical value.

So, we use a kind of human learning method, it is very efficient, and can achieve small samples, cumulative learning, real-time updates. It does not modify the parameters of "old knowledge", so there will be no "catastrophic forgetting" problem. It does not require the BP gradient optimization process, so its computational amount is consistent with the inference process of the large model.

5 we implemented the three conditions proposed by Professor Yann Lecun

With the three deep learning giants and Turing Award winners, Professor Yann LeCun believes that the right direction of AGI is the "world model" and the road is to achieve "humanoid AI". They proposed three conditions:

(1): Need a world model. These include demand modules that need to model basic needs such as happiness and hunger, as well as value modules that predict value.

(2): Need a logical reasoning ability compatible with the neural network.(The current reasoning ability is based on the plug-in symbolic reasoning).

(3): Need a "general decision-making ability", can top down, decompose decisions. Can't be intensively trained a million times for every task!

Although they put forward these ideas, they have no complete technical solution. And in our scheme, we can achieve the above three conditions.

5.1 We built the world model

The input Tokens activates the Tokens combination in memory, and the high activation value Tokens combination is the activated "world model" (some Tokens may already appear in the input and other Tokens may not yet appear in the input). Then, according to the predicted decision process of "seeking benefits and avoiding harm", decide whether to further confirm the existence of other "high activation value Tokens", which is "pattern recognition". The world model is "common sense", which is the Tokens combination mode composed of subjective Tokens and objective Tokens, such as "demand", "reward and punishment" and "emotion". Humans use "common sense" to "pattern recognize" things.

After each new information input, the machine needs to conduct a chain association activation, and then store the Tokens as "simultaneous storage" mode. Simultaneous storage is the use of a mechanism to reflect the time interval between Tokens. For example, the time interval can be determined according to the closer the Tokens of the time is approaching, or the closer the storage location is approaching, or according to the time information brought by each Tokens.

Every time it gets a new Tokens, the machine needs to have a more updated activation value to find a way to achieve the reward and avoid the punishment. The set of these paths is the overall response path. The overall response path may be a network-like structure, and many local paths may lead to both reward and punishment symbols.

Because of the activation value transfer path to the reward symbol (or penalty symbol), that is, we realize the advance and step of the reward and penalty function. Therefore, we have solved the problem of sparse and lagging reward function in the current reinforcement learning process. The machine can find the initial optimal response path by a search process similar to AlphaGo.

If the overall reward and penalty value does not enter the acceptable preset value (or no convergence), the machine cannot decide whether to choose or exclude certain specific paths, so as to maximize the benefits. The machine needs to further identify the input information and add more Tokens to subdivide certain reward and penalty activation value transfer paths, so as to further help the machine to select or exclude certain specific paths. This step is the process that machines create spontaneously and actively find information to help them make decisions. This process proceeds iteratively until the reward and penalty statistics reach the accepted preset values or converge.

When further identifying the input information, the high activation value Tokens is either because of their high memory value, such as the representative of a class of things Tokens, or the Tokens closely related to the input Tokens, such as similar, or often near the appearance. Therefore, the Tokens combination of high activation value being activated in memory is the representative Tokens combination related to the input information. These representative Tokens combinations are the "world model" temporarily created by the machine, which we call the "expectation model". It is both a summary of past experience (Tokens memory value after survival of the fittest) and directly related to the current specific input. It is temporarily created through high activation values and is the "expected model" of the machine for the current input Tokens combinations.

The spatial or temporal relationship between the Tokens in the machine refers expected model already present in the input and the Tokens not present in the input, Given the temporal and spatial location of Tokens, Predict the temporal or spatial location of those Tokens that have not yet appeared; These high activation values Tokens, which have not yet appear in the expected model, Is the expected Tokens; the machine assigns the time and space location of the machine's sensor search according to the time, space, and size of the expected Tokens in the expected model, And determine the type of sensor used based on the expected Tokens properties (e. g., speech, image, or touch), And determine the resolution to be used based on the properties of the expected Tokens (such as the size). This is the machine's "on-demand identification" process. This process can be performed iteratively.

Selective attention is used to extract Tokens from the input information, and the machine extracts Tokens from the input information according to the recognition interval and resolution given by the selective attention recognition. In this way, the problem of wireless granularity of image information can be solved (the machine extracts the information in the image on demand). When the machine extracts the specific interval data in, it preferentially extracts the overall topology, shape outline, main lines and main texture Tokens in the selected interval in the way of overall feature first. Then, the machine obtains the relevant memory in the memory network through the chain association activation process, and combines these memories into different weight expectation models according to the weights.

The machine uses the decision process to determine whether to further identify the input information according to the activated reward and penalty Tokens (the activated value of the reward and penalty Tokens, which is the expected reward and penalty value), or whether to respond to the input information.

If the machine decides to further identify the input information, the machine further extracts the "expected Tokens" from the input information by imitating the relevant experience of obtaining the "expected Tokens" in the past. Therefore, the machine is the Tokens that constantly iteratively extracts the input information through the attention mechanism, and each extraction process may use different sensors, with different resolutions, for different recognition intervals. So for the same input thing, the machine may extract different types, different intervals and different resolutions of Tokens, and use the combination of these Tokens to form a "hierarchical representation" of the same thing."Hierarchical representation" refers to the Tokens that extracts information step by time in the overall way of low resolution in the interval.

The high activation value Tokens is used to form the expected model; its theoretical basis is that these high activation value Tokens come from two parts: one is the common features of similar things; because common features are widely found in similar things, they are highly repetitive, so they are usually high memory value Tokens. Therefore, in our scheme, the machine is to first identify large categories (obtain abstract concepts) through common features, and then gradually add more Tokens to limit the scope (from abstract concepts to concrete concepts).

Another source of high activation values is that there are similar Tokens in the input Tokens and in a specific memory. These specific Tokens, which are directly activated by Tokens in memory because of similarity activation, and other high memory values Tokens with its proximity relationship are also prone to higher activation values. Because of the short activation path, so in the relational network, special Tokens activates a specific "expected model", a way to quickly locate the expected model through special Tokens.

Therefore, the identification process of input information is to identify which large category it belongs to through common features, and then to determine which specific subcategory it belongs to through unique features. The machine iteratively increases the Tokens for identification through selective attention. In this process, the previously activated Tokens, whose activation value fades over time. If they are re-activated by the newly input Tokens, their activation values are consistently maintained. If they are unrelated to the new input Tokens, their activation values slowly fade away and gradually exit the decision process.

The "world model" contains two aspects: 1. The machine knows the world iteratively in the way of "pattern recognition". Machines know the world in the way of "pros and cons". This is because "pros and cons value" is the core "world model" established by human beings. It is a "world model" that guides all human behavior.

So, we implemented the "world model."

5.2 We achieve logical reasoning capabilities that are compatible with neural networks

All the "reward and punishment" Tokens motivated by the input Tokens, their activation value size is the value prediction.

The propagation path from the input Tokens to the activated reward and penalty Tokens is the reasoning ability that is fully compatible with connectionism! A memory network is a neural network organized by Tokens to transfer according to activation values. The essence of activation value transfer is the inference process that realizes the attention mechanism.

Tokens Combination in each Token, through chain association activation process, activation and their common Tokens combination, through the activation value accumulation process, can realize from the input combination (known N Tokens specific combination probability), and the most relevant Tokens combination (M Tokens specific combination probability), and the final activation value distribution in the memory library is the Bayesian inference results.

In fact, the attention mechanism in large models has achieved logical reasoning capabilities compatible with neural networks. However, there are two defects: 1. Deep learning destroys the original organizational form of Tokens time and space, leading to knowledge being difficult to be understood and imitated.2, the lack of "subjective Tokens" (such as needs, emotions, and pros and cons). So the inference process of the big model is flawed.

Turing Award winner Professor Yushua Bengio, one of the big three deep learning, believes the most important step in general AI is to combine neural networks with causal reasoning. In fact, our scheme has achieved this: the memory network is a fully connected neural network, from the input Tokens to the activated "world model", is the causal reasoning of the objective world organization; from the input Tokens to the input Tokens to the activated "subjective Tokens combination" (representing the demand, emotion and rewards and punishment Tokens), is the causal reasoning between the objective world and the needs of the machine itself.

So, we implemented "combining neural networks and causal inference."In fact, the current large models have realized the objective reasoning ability and some subjective reasoning ability, but their reasoning process is difficult for humans to understand and imitate, so it is difficult to use.

5.3 We achieve a hierarchical "general decision-making capability"

The machine only reinforcement learning one task: " How do you meet your needs?"And only deal with one task" how to meet your own needs "? So our machine, the decision is "to face their own needs", while currently other AI solutions, the decision is to face all kinds of "task itself".

Information input, produce all kinds of associations, there are good and bad. Reduce the probability of Tokens that brings "punishment", and increase the probability of Tokens that brings "reward". This is the "general decision"! This is similar to human decision-making, so universal!

With the advance and step of the reward function, the machine has the "decision-making ability". With the general goal of "seeking advantages and avoiding disadvantages", machines can achieve "general decision-making" capabilities.

5.3.1 The current "machine learning" is not really "machine learning"

In the face of a new task, people predict the "good or bad" of different decisions based on their own experience, and choose at most a few schemes to try. For a new task, machines currently rely on reinforcement learning to "keep trying," or (1) try a million times to see the results (Google AI for various games), or (2) humans tell me good or bad (GPT-4, big model, RLHF)[23], Then gaining knowledge of decision to deal with this issue.

Therefore, the current machine learning takes the way of "try first" + "then eliminate". So they should call it "machine evolution," not "machine learning."So we proposed that AGI requires real "machine learning." What is the real "machine learning"? In our opinion, real machine learning should be like human beings, facing a new task, which can predict the "good or bad" in different decision paths according to their own past experience. By choosing a limited few solutions to try, we can obtain the decision knowledge to deal with the new task. Furthermore, we believe that real learning should also be similar to children's learning style, by directly acquiring the accumulated human experience through language. In the face of a new task, an attempt is not needed, a direct success! For example, in the laboratory, when the teachers teach the children to do experiments, they directly pass on the existing human decision-making experience to the children through language teaching. After the children can obtain the knowledge conveyed by the teacher, they can use the experience gained by the language and interact with the environment to directly complete the experiment. Although it may be the first time that the children do these experiments!

Real tasks vary greatly, and real scenes vary greatly, and human beings can not put each type of task into a large number of scenes to "reinforcement learning"! So, you must change your thinking! The idea is to turn all tasks into a single

task: "How do you meet your needs"? All the training process of the machine is about training this task. So, facing this task, the machine has a lot of "state" and "policy knowledge, so it can predict the potential" pros and cons "estimates of" different decisions ".

The task given to machines is the background information of the task of "how to meet their needs". If "gaining human recognition" is one of the needs of machines, then machines will incorporate "completing human tasks" into the overall pros and cons statistics in the pursuit of "meeting their own needs". This is similar to the human, facing the boss, you will weigh the pros and cons to make different decisions. For example, one decision you make may be to proactively find more information to analyze the pros and cons of the task before making the decision. And actively looking for more information is the new task of by themselves. If the machine makes the same decision, then it is the machine assigning itself tasks, that is, the machine programming itself. In fact, our machine uses this decision-making process: weighing the pros and cons, and having the potential to actively find information to help you maximize your benefits.

5.3.2 How to achieve true "machine learning"?

Ten years ago, we thought that to create real "knowledge", we should start from the perspective of information statistics. Unlike "deep learning", we believe that machines should learn in accordance with the human learning model, using small samples and knowledge accumulation. Therefore, at the beginning, it also tried to use "symbolic expression", "causal logic" and "knowledge network".

After a few years of trying, I found that the first step of the road was blocked. Because "symbol expression" "dog" how expression? Need to pick out all of the characteristics of the "dog". But a "dog" can be either an animal or a person! It can be "a celebrated character" or "a despised character", where the meaning of the symbol "dog" varies greatly in different contexts. So the essence of "dog" is the sum of the relationship between "dog" and all other things. So the "dog" must be put into the whole knowledge network, defined by its relationship to all the other knowledge. So, "symbolism" doesn't work! Because "dogs" can't be separated from other knowledge! A "fully connected knowledge network" similar to deep learning, which is our first conclusion.

Because the "dog" must be placed into the entire knowledge network, defined by its relationship to all the other knowledge. So you must have enough knowledge to understand the "dog" thing. Therefore, the amount of knowledge must be sufficient, so that through enough background knowledge to understand what a dog is. This is our second conclusion.

When we look back, isn't that what the big model does?"Deep learning" is to do a fully connected network, and the big model is to do "use a lot of knowledge to build a fully connected knowledge network". So why don't we see robots walking around the streets? Because only the knowledge network is not good! The machine must also be able to "interact with the environment to make decisions"! Studies have shown that humans make more than 30,000 decisions a day. Beyond the industry, what allows the machine to make its own decisions is only reinforcement learning algorithms. So, one possible way to general AI is: big model + reinforcement learning algorithms. In fact, GPT-4 has already achieved "full knowledge + fully connected network + RLHF", and RLHF is reinforcement learning. Google Published the Gato model in 2022, and has taken the road of "all knowledge + fully connected network + reinforcement learning".

So why don't we see Google with robots walking on the streets?

The core obstacle to this path is the reinforcement learning algorithm, the two prerequisites required[24]: (1), The machine needs to know the reward information it can get at different decision paths. Because reward information is scarce and delayed, the current problem is a lot of trial and error training.(2), the machine needs to search for all possible decisions.

These two conditions, can be perfectly satisfied in the game. The game can constantly try, the decision search space has boundaries (can also be trimmed to reduce the search space). But in real life, there are many problems without constant trial and error (such as taking care of children, no one wants to let you keep trying!), There is no clear boundary, so this problem cannot be solved! That's why Google keeps launching AI that can play a variety of very complex strategy games, but has been unable to launch the most basic "home nanny robot"! In fact, in daily life, the vast majority of decisions are far less complicated than the decisions in games! But because in real life, a lot of things can not be massive trial and error! And in real life, there are no clear boundaries to the relevant information. So the two difficulties above, leading to Open-AI or Google, through "big model + reinforcement learning", can only be used for things that can be massive trial and error. So AIGC, from AGI, is still a long way off!

Our decision-making plan is also essentially reinforcement learning, but only reinforcement learning how to seek advantages and avoid disadvantages. And we took advantage of the chain association activation process, automatically limited the search range! Search only for the activated information! Moreover, we use the logical chain of "Tokens" and "reward and punishment symbol" to automatically predict the reward and punishment information, rather than

only post-hoc feedback to obtain the reward and punishment information. So we perfectly solved the problem that Google's decision-making AI can only play games! This is because we have simultaneously realized the "objective common sense" + "subjective common sense". In the existing technical route, the technical route is to realize "objective common sense" first, and then to establish "subjective common sense" through "RLHF". So the current technical route, "subjective common sense" is obtained through post-hoc feedback, so it can only apply to areas where there is a lot of trial and error.

5.3.3 Implementation process of "universal decision"

The machine is in any environment, and the input information includes all the sensor information. So at any moment, the environmental information that the machine is in is a part of the input information.

Machine and environment interactive decision-making, including two aspects:

1, and the choice of the optimal decision-making.

2. Execution of the decision-making process.

These two steps are not separate! Is intertwined, parallel processing!

The first question that the "universal decision" needs to solve is: What is the reward function? Inside the GPT-4, and within the A lpha go, the reward comes from the final external feedback. In our AGI, the reward comes from the "reward" and "punishment" symbols activated by external information, and the size is their activation value.

Step 1: What is the purpose?

When the information is input (outside + machine monitoring information) input, some reward and punishment symbols are activated.

Each transfer path from the activation value of the input reward symbols and punishment symbols is a potential logical link for generating a reward or punishment.

If each underlying feature is truly realized on this logical link, then the reward or punishment spread by this logical link is also realized.

Therefore, the response of the machine to any input information is the same: increase the probability of reward logic chain occurrence, reduce the probability of punishment logic chain occurrence, to achieve the purpose of seeking benefits and avoiding harm.

Step 2: How to plan with a purpose?

1. How to increase the reward link and reduce the occurrence probability of the punishment link?

The Way is increasing, or decreasing, the realization probability of the high activation value Tokens combination on the link. The high activation value Tokens combination on the link is the high-weight Tokens combination of this link. When they are true, the activation value spread along this link is true, so the final activated reward, or punishment, is also true.

2. How to operate it specifically?

From the transfer path of the activation value of the input information reward and penalty symbol, the N Tokens with the highest activation value is selected, which causes the reward or brings the real top implementation path. The goal of the machine is: 1, to let the Tokens implementation on the reward path (which is to imitate the past experience and let them appear in the input information).2. Make the Tokens on the penalty path not possible (that is, imitate past experience and avoid them from appearing in the input information).

Therefore, from the logical pathway of input reward and penalty, select the N Tokens with the highest activation value and contain the propagation path of the activation value, which is the top implementation path. Why does the machine only select the N Tokens with the highest activation value? Because these Tokens, either because they are representative Tokens of things, have high memory values and obtain higher activation values; or Tokens closely related to the input information. Due to the small number, equivalent to less attribute restrictions, so the concepts most closely related to them are usually "abstract concepts".

Due to the frequent use of language symbols, the language Tokens often obtains a high activation value, becoming the core Tokens with the highest activation value constituting the combination of "abstract concepts" Tokens, making the language symbols become the representative of the concept itself. Such as "eating", "escape" and other abstract concepts. It should be pointed out that "abstract concepts" are not the patent of linguistic symbols, and animals can also have "top-level decisions".

So the process of machine decision-making is to prioritize "abstract concepts" and then gradually add more Tokens to form a more concrete combination of concepts. This is the top-down, gradual process of decision-making and execution. We call this process "segmented imitation".

Specific example of the segmented imitation method:

Consider the set of input Tokens as A and the set of response Tokens as B; the machine looks for the activation process through A and B chain association for high activation values Tokens. These Tokens are Tokens that are connected to both A and B, because they obtain activation values from both A and B. They are the middle bridge Tokens that connects A and B. This process proceeds iteratively, enabling top-down, layer-by-layer decisions.

How to do it in a computer? The methods adopted are as follows:

(1) the external input Tokens to determine the activation value of the reward and penalty symbol (the Tokens exceeding the preset value as the target), and establish the level target.

(2) Starting from the reward and penalty symbol with the highest activation value, find the N Tokens with the highest activation value on the transfer path of the activation value from the input to each first-level target, which are the logical link to realize the corresponding reward and penalty. The Tokens on the link is the secondary target.

(3) The machine takes each secondary target as the new target, takes them as a new input Tokens, gives them the initial activation value, and initiates the chain association activation process again. So, the Tokens with the highest activation value are the Tokens combinations associated with the external input Tokens and the secondary target Tokens. This is because we employed activation value accumulation and activation value extinction, and only the Tokens associated with the most recent input Tokens can maintain the activation state. So these Tokens are the level 3 goals.

(4) This process is performed iteratively, and the machine can break down each level goal into hierarchical logical links to achieve them.

(5) For each expansion of the decision-making process, different reward values or punishment values will be selected to enter the accumulation. According to the principle of seeking benefits and avoiding disadvantages, the machine chooses the subpath that brings the reward value and avoids the sub-path that brings the penalty, thus increasing the cumulative reward value. When the machine finds that the total reward and penalty value converges, that is, it cannot be further improved, that is, the benefit is maximized. The machine stops expanding further and enters the execution process. This is the hierarchical "general decision ability" proposed by the Yann Lecun tutorial, and also the logical reasoning ability proposed by Professor Bengio that is compatible with neural networks.

Why are only the N Tokens with the highest activation values selected for each expansion? This is because past experience is impossible to fully match the current and reality, so by selecting only the highest activation value Tokens, it means that the "model" is either abstract (widely applicable) or closely related to the input Tokens (good match). The purpose of selecting only the Tokens with the N highest activation values is to achieve empirical generalization. So, in our scheme, the empirical generalization is implemented automatically.

For example, the machine has use hammer nail experience, in need to hit nails, and no hammer, and the input Tokens stone, in order to achieve primary goal (reward symbol or punishment symbol, complete the task, reward, or avoid punishment), in the activated logic link, may contain the Tokens combination represents the hammer. Then, these Tokens combinations are the secondary target.

Machine according to the chain of memory association activation process, may be found the M hammer target activation value transfer path, may be from the "memory toolbox", may also be from "to teammates borrow experience", the activation value transfer path is to improve "hammer" Tokens implementation probability path, which is the secondary path to reward.

Since the stone-related Tokens appears in the input, the total Tokens of the hammer and the stone (such as weight data, size, hardness sensation, etc.) is likely to obtain a higher cumulative activation value, which can be selected as the first N high activation values. They become bridge Tokens, making the stone-related Tokens a secondary path to rewards. This is the empirical generalization process, through the Tokens shared by the stone and the hammer, allowing the Tokens of the stone to transfer activation values to the reward symbol. The reason for this is that "stone" and "nail hammer" have some common attributes (there are Tokens, and this part of Tokens can be repeated in various kinds of "hitting nails with nail hammer" scenes), and they are the bridge of experience generalization. It can be seen that, in our scheme, the empirical generalization process is done automatically.

So, which secondary path does the machine take to the rewards? At this time, the machine needs to activate the Tokens space updated according to the new chain association activation process, and choose its own decision-making path again according to the principle of seeking advantages and avoiding disadvantages. Some paths, which may bring both

rewards and punishment, may make it difficult for the machine to calculate the rewards and penalties. If the machine finds that the reward and penalty value statistics do not converge, the machine's decision is to further identify the information to converge each reward and penalty value transfer path.

For example, "toolbox in memory" needs to confirm the probability of "toolbox" currently appearing in the input (implementation)? This probability can further converge the reward and penalty value of this path. At this point, confirming the probability that the "toolbox" is currently present in the input becomes a new goal for the machine to create itself. To achieve this "new goal," the machine needs to imitate past experience to execute it. If in the past memory, its toolbox is hanging in the waist, then it imitates the past experience to determine that the toolbox related Tokens appears in the input, the most likely imitation process is to use the "hand" to pat the waist, to reproduce the past hand touched the "toolbox" of the various sensor data combination. Because this decision costs the least power and uses the least time, and can maximize the benefits of the machine itself, this is the preferred decision path.

For another example, the "borrow from a teammate" path requires increasing the implementation probability of the Tokens in this path to transfer a greater activation value (to get a greater reward) to the reward symbol. So, the most likely experience a machine imitates is to look around, or ask.

Therefore, in our scheme, the decision of the machine is very complex. In a decision path, it may be nested W decision and execution processes, but at any time, the only goal of the machine is to "seek advantages and avoid disadvantages". All decisions are derived around this goal. Therefore, the decision of the machine is very flexible, it always changes according to the state of the environment, and there is no preset process. The only preset process is just: "Seek the best and avoid the bad."

The above process proceeds iteratively, and each time a new reward and penalty symbol is activated. The machine counts the activation values of these reward and penalty symbols until the activation values of the reward and penalty symbols converge. Then the machine establishes the optimal response path.

It is possible that the machine will make decisions in response to input information or to find more information to continue making decisions. Either way, machines increase or reduce the implementation probability of a particular Tokens by mimicking past experience. At any time, with the new information input, the new information will update the activation value distribution in the memory bank through the chain association activation process. At this time, the machine needs to re-count the reward and punishment information according to the new state, and re-find the optimal decision. Only with new information, the process continues all the time.

Step 3: With the planning, how do you implement it? Execution is improved by mimicking past experience, or by reducing the probability of Tokens.

1. Choose a small number of underlying features with the highest activation value to abstract the decision path.

2, adding more high activation value of the underlying feature abstract decision path embodiment.

3, steps 1 and 2 above proceed iteratively until the decision is decomposed to the drive command that can be executed. Drive command: send waveform to the speaker, send drive command to the motor, send display data to the display screen, and send set parameters to the facial expression display system, etc.

4. New input information may be encountered at any time, and the new input information will change the activation value in the memory bank and change the reward and penalty situation, so the machine may change the original plan at any time in the process of implementing the optimal response path!

Step 4: segmented imitation during decision making and execution.

The machine can find the experience related with the current input through the chain association activation process. Among these experiences, the probability of a small number of high activation values Tokens is usually representative abstractions due to their high abstraction. These experiences contain "antecedents" and "consequences" related with the input Tokens, which are the objects of empirical generalization.

The essence of empirical generalization is to use the effect of the existing process to achieve the effect of the unfinished process. In our scheme, it is completed automatically completed by the transfer of activation value of "common Tokens" in the two processes. Since the Tokens is not consistent in the two processes, this corresponds to the mismatch problem in the empirical generalization process. But this problem, in our scheme, the experience of the two processes, is through the common Tokens to achieve the activation value transfer process, to automatically complete the generalization.

It should be particularly noted that the concept of machines is formed by various Tokens through large stereoscopic networks. The same Tokens may be distributed in different memory segments. These Tokens are likely to come both from their own experiences and from input to linguistic symbols.

Therefore, when the language symbol itself is activated, the relevant Tokens represented by the language symbol are activated. However, the language symbol itself has the sequence, and the language sequence usually contains the expression Tokens combination sequence, so behind the language symbol sequence is the Tokens temporal and spatial information flow. Their temporal and spatial combination order is the "causal relationship". Moreover, these "causal relationships" can form a close activation value transfer relationship through the chain association activation process of "language symbols". This close activation value transfer relationship, which is itself a kind of "experience". So, in our scheme, the experience comes not only from the machine itself, but also from the "experience of others" gained through linguistic symbols. Therefore, our machine can not only learn "experience" through language symbols, but also imitate "experience" through the Tokens information flow composed of language symbols.

6 our scheme and the current large model road

Our solution solves the following problems:

6.1 How to "build up common sense" problem

Deep learning destroys the time and space relationship of the original Tokens! In our scheme, the "chain association activation process + memory and forgetting mechanism" is used to realize the attention mechanism. However, we did not adopt deep learning, so our scheme, the knowledge created, retains the original temporal and spatial relationship of Tokens. And the original Tokens combination way, it is exactly the basis of the human "concept". So, in our scheme, the "knowledge" it creates is the knowledge that humans can understand and imitate.

In our scheme, the essence of "knowledge" is the permutation relationship of Tokens in time and space, and the prediction of different Tokens permutations of the agent. And Tokens in time and space arrangement nature is "causal", the Tokens in time and space is not simple near time, space, but agent, can repeat the relationship, they actually span the span of time and space is likely very big, but through the chain of association activation process, the time and space span large Tokens, formed a close activation value transfer relationship, this is the knowledge. If knowledge includes Tokens related to "needs", "emotions," and "pros and cons", this can predict potential pros and cons, so the arrangement of Tokens represents "knowledge". The common permutations are "common sense."

6.2 The question of whether the machine can be conscious "

We solved how to give "self-needs to a machine."Therefore, machines can make independent decisions, have self-evolution, can have their own emotions, and can pursue "self-needs", so our machines are "conscious".

6.3 The "universal decision-making" problem

Facing any task, the machine makes decisions according to "seeking good and avoiding harm". The task given by humans is a by-product of machines' pursuit of "self-needs".

This is the same thing as completing the task your boss assigned. You are also in the pursuit of "self needs", to complete the tasks assigned by the boss. If there is a conflict between the two, you will make a variety of flexible decisions according to seek the best and avoid harm, test the boss's true intentions and consider the boss's bottom line, so your decision will be very flexible!

6.4 The "language understanding" problem

Because we didn't break the original time and space relationship of Tokens. The Tokens temporal and spatial sequence represented by language sequences can be understood and can be imitated. So machines can learn various skills directly through language, just like humans. Read the oven manual and start the toast[5][6][7].

6.5 We think that our path is a viable way leading to the AGI

6.5.1 Advantage 1: Ability to handle tasks that fail a lot of trial and error

Such as autonomous driving, home nanny, taking care of the elderly, accompany children, engaged in "workers, peasants, soldiers and business".

Because we are "humanoid" AI, we can make general decisions and learn skills in language! And the current big model can't handle these things!

6.5.2 Advantage 2: Can solve the "illusion" problem

Large models have only "common words" obtained from local statistics and no factual memory.

Our scheme, first, stores memory and then extracts common information from memory. So we have our own "fact database", and it is integrated with knowledge.

6.5.3 Advantage 3: Ability to learn skills directly through the language and imitate them

Because we do not destroy the temporal and spatial relationship of Tokens combination, the spatiotemporal relationship of Tokens represented by language can be understood and imitated! This point, no matter now, or in the future, the big model can not achieve! For example, on the first day at a bakery, it will ask the boss for an "oven" manual. Read it and start baking without individual training!

6.5.4 Advantage 4: Be safer

(1) At present, artificial intelligence is a single goal. From the perspective of decision making, it is a "one-minded thinking" artificial intelligence "to achieve the goal". Such an artificial intelligence, it doesn't think about anything outside the goal, the decision is still a black box. Think how dangerous it is if "stuffy pot" + "one-track mind" people take control of your life! If such artificial intelligence is allowed to fully control human life, it may completely bring incalculable disasters to human beings because of the wrong understanding.

(2) In our scheme, the "demand type" of the machine can be preset, the values can be trained, and the human values can be aligned. At any time, the machine will consider various goals, and there will be no "extreme" behavior. Moreover, in our scheme, the decision is visible, modifiable, and "white box".

7 The underlying logic of our scheme

7.1 The chain association activation process is the attention mechanism

First, we believe that the nature of knowledge is information. And human knowledge, is a very small part of the information. This is because we humans have a limited resolution of the information. The relative spatiotemporal relationship between the arrangement of A and B atoms on A grass is also A kind of information, but we will not identify it.

So in the process of evolution, humans have developed the Tokens recognition ability. Tokens Is the smallest information unit commonly used by humans, such as a straight line. Tokens itself is the "world model", it is the smallest "world model" for human beings to build a magnificent palace of knowledge. In the process of evolution, humans have formed the "pattern recognition" ability to adopt "models" such as Tokens to identify the surrounding information, which greatly improves the energy efficiency ratio of information recognition. It's a gift from evolution.

If we arrange the "Tokens" of everything from the "Big Bang" to the "present" in order of space and time. We just get an information tensor. It is all the knowledge that man has. Faced with such a treasure house of knowledge, if our agents outside the universe want to know about it, they will make statistics on these Tokens.

The first question: "How many independent Token's do we have"? In our scheme, the similarity relationships answer this question. Second question: "the quantity distribution of each Token"? In our scheme, repetitive relationships answer this question. The third question: "How are the Tokens arranged"? In our scheme, proximity relationship answers this question. We can see that in our scheme, through the chain association activation process, memory and forgetting mechanism, is to make a statistical description of information!

In the attention mechanism of large models, the Tokens combination correlation is inferred by the pairwise correlation between Tokens. Then again to speculate a larger Tokens combination correlation by pairwise correlation. This process takes multiple iterations to obtain the correlation of the different Tokens combinations to each other. The pre-training process is to find the correct "optimal coordinate base" through the trial and error method (deep learning). With the help of the attention mechanism, the obtained "optimal coordinate base" is only for the "common information". The essence of this process is a Bayesian inference process: the conditional probability of a particular Tokens by partially known probability.

In our scheme, the correlation between Tokens, is obtained by induction. The chain-type associative activation process is to use the correlation obtained by pre-training (partially known probability) to obtain the conditional probability that a certain Tokens may appear. The chain association activation process is to find correlation (the reasoning process of attention mechanism); and memory and forgetting mechanism are induction.

7.2 The core of the attention mechanism is to create "common sense"

Knowledge is the arrangement of Tokens, and common sense is the common way that Tokens is arranged[16]. The big model of the core problem is that the human knowledge (Tokens arrangement), converted to its own system of knowledge (because of the deep learning destroyed the original Tokens space-time relationship, lead to big model of knowledge, human is difficult to understand, unable to imitate), it uses its own knowledge system to solve the problem, and then translated to human. Therefore, deep learning destroys the spatial and temporal relationship of the original Tokens, which means that it destroys the original and human understandable organizational form of Tokens, and transforms it into the organizational form that machines can understand. From a machine's perspective, it retains the way Tokens is organized because it correctly finds "common information."But from a human point of view, the knowledge that it produces, and the knowledge created by human beings, cannot be directly interconnected, or can not be directly borrowed from each other.

Because two sets of system of the underlying language can't communicate with each other, so it is difficult to give the machine "innate knowledge" (such as innate requirements, innate reward and punishment function and innate emotion function), so can only remedy the way, the RLHF, or using pl-in knowledge base, to solve part of the problem, and only through the "yes" or "No to communicate, the robot, can only be a" scripted "" nerd ", not really flexible to solve the problem.

Therefore, in our scheme, the most core is to establish "common sense" without destroying the original form of time and space organization of Tokens, and need to include the "subjective common sense" of the machine.

In order to establish "common sense" under the original form of time and space organization of Tokens, we adopted information Tokens, retain time and space information storage, and adopted the chain association activation process, and adopted the memory and forgetting mechanism to realize the induction of the chain activation value transfer relationship between Tokens. At the same time, we imitate the organization form of "common sense" and preset the Tokens combination representing the innate demand, the innate reward and punishment function and the innate emotion function. Then let the machine make independent decisions and evolve itself according to the principle of seeking benefits and avoiding disadvantages, and constantly expand the memory bank around the innate knowledge to form the whole knowledge network, so as to create "objective common sense" and "subjective common sense".

7.3 We can accomplish only one thing: "Create common sense"

In order to "create common sense", we need to solve (1) "give the machine self needs".

In order to solve (1), (2) "how to create understandable knowledge" problem.

In order to solve (2), the problem of "how to create a fully connected knowledge network without using deep learning" should be solved. Then, subjective Tokens and objective Tokens can be realized through attention mechanism. This is common sense.

The establishment of a relationship between subjective Tokens and objective Tokens is the "front" + "step" of the excitation function, which can realize the "general decision-making ability" by "seeking advantages and avoiding disadvantages". Driven by "self-needs", machines can achieve "self-evolution".

7.4 We establish an infant AI

"Build a baby machine, then learn lifelong, and grow yourself". The idea has been around for years, but we were the first team to propose detailed solution steps.

8 A simple example

Below, we illustrate how the machine makes decisions and responds with an example.

Background: Lao Wang went to other places for vacation, took an assistant robot, stayed in a hotel room...

Lao Wang: " Hello...".

Robot: There are many Tokens activated in the memory bank, but in these activated Tokens, there is no reward symbol whose activation value exceeds A1 (A1 is a preset threshold), and no activation value exceeds P1 (P1 is a preset threshold).

It is constantly receiving the external information and internal information from the sensor, and uses low resolution priority to extract the Tokens in these information, stored in the memory bank. According to the same process, the initial

activation value is given to these Tokens. Since there is no reward symbol / punishment symbol with high activation value, the activation value given to these Tokens is relatively low according to the predetermined procedure. Therefore, in the subsequent chain association activation process, the propagation range of the activation value is very small, and the chain activation process is very quickly completed.

The machine starts updating the memory value. Because activated Tokens obtain low activation values (due to low initial activation values and low activation value spread), their increased memory values are small and a lot of their information is forgotten in a short time. At the same time, because the activation values of the reward symbol and punishment symbols in the memory bank are relatively low, that is, the potential reward and the potential punishment are relatively small. So the best decision path formed by the machine is to continue receiving information. This is because giving power itself is a punishment, and if you don't get a reward, then the optimal decision is not to waste power.

After each chain association activation process, the machine needs to check whether there are reward or punishment symbols whose activation value exceeds the preset threshold. In this case, the optimal response formed by the machine is to extract the Tokens from this information with low resolution and save it in the memory bank. Following the same process, the above process cycle proceeds.

Suddenly, the audio processing system introduced a series of audio Tokens (still extracted at low resolution), and these Tokens, following the same process, were given a lower initial activation value and underwent a chain association activation process. This input Tokens, some Tokens in the process of chain propagation, because of similarity, activated many similar Tokens in the memory library, there is a close relationship between these Tokens and many reward, punishment symbols, so the activation value chain propagation process, there are a lot of reward and punishment symbols are activated.(These Tokens are usually the owner's voice print features, such as their unique timbre).

Because this time, many rewards, punishment symbols get more than the preset activation value. Assuming N reward symbols and M punishment symbols have the activation values above the preset value. The machine targets both N reward symbols and M punishment symbols, so that the machine autonomously establishes N + M targets at the same time. So, in our scheme, the goal is machine-autonomously generated, is multi-target generated at the same time, rather than artificially preset a total reward function. In our scheme, all the response of the machine is based on the principle of seeking advantages and avoiding disadvantages.

Therefore, after the machine creates N + M targets, the machine plans its own response path principle is to increase the probability of the activation value of the reward symbol and reduce the probability of the activation value of the penalty symbol. So the machine's decision is all around achieving the reward and avoiding the punishment.

The machine first processes the reward / penalty Tokens with the highest activation value, which may be one or more penalty Tokens; in the memory bank, the propagation path of transmitting the activation value to the penalty Tokens may be the underlying feature input of the voice print of the owner, for many of the owner; and the activation value further transmits the activation value in the memory bank.

Of these memories, one penalty, Tokens, had high activation values. The Tokens that can obtain a high activation value is nothing more than several cases: (1) the penalty Tokens has a very high memory value. One possible reason is that when it is stored, its activation value is high, while the memory value increment and the activation value are positively correlated. Another reason is that it is often activated, by repeating high memory values.(2) Multiple input Tokens's pass the activation value to this penalty Tokens through different pathways. Such as the host "tone Tokens", "word Tokens", "master state Tokens", "host expression Tokens", "current environment related Tokens", if these Tokens and similar punishment symbols in memory, so they complete the activation value chain propagation process, and they are related to Tokens may obtain high activation value.(3) There is a tight activation value transfer relationship between this penalty Tokens and a specific input Tokens. That is, they always appear in memory. Therefore, they form a "proximity relationship" and a "high memory value relationship", and the propagation path is very short, and the activation value transfer coefficient is high. Therefore, the attention mechanism may not only use comprehensive reasoning (such as multiple Tokens to specific reward and punishment symbols, comprehensive experience), but also use special case reasoning (such as the specific activation value closely transfer path, the specific experience). The penalty Tokens is high and may also come from the activation value distribution established by the previous Tokens input. Although the activation value of the high activation value Tokens fades over time, if the activation value is high enough, it will influence the decision of the machine for a longer time. This is very similar to humans.

In this example, the propagation network composed of activation-value propagation paths contains a very large number of Tokens that it is difficult to formulate. But it is usually the language symbols that have the highest activated value (because they are the most commonly used and have the highest memory value), and if they are combined in their space-time order, the main idea may be "don't lie down (the cause), be scolded by the owner, very sad (consequences)".

The machine immediately starts searching for the optimal response path to avoid the probability of this penalty symbol occurring. The principle of machine decision making is to increase the probability of reward symbol and reduce the probability; reduce the probability of the path of the activation value and increase the probability of the path of the activation value to increase and reduce the probability? Each concept is a local tight network in the memory bank, and the machine needs to reduce the probability of high activation value Tokens in this local tight network, thus reducing the occurrence probability of this reward and penalty logic link.

For example, in the case of the memory of the machine, when the owner "uses a similar Tokens reprimand", the memory stored its internal sensor data and the external sensor data of the time; some of the Tokens were forgotten because they did not repeat again and did not obtain the enhanced memory. But the Tokens combinations that can repeat with this "punishment symbol" are "lying" related Tokens, and some "specific time Tokens" and "specific occasion Tokens", which obtain higher memory values because of their repeatability. And because it is a recurring combination, each other pushes up the activation value, so to obtain a far higher than the repetitive memory value. And because they can repeat, their combination achieves higher activation values each time, so they are easier to be activated and easier to remember, so it's a positive cycle process. This is the process of summarizing the experience.

If once, in a similar environment, the owner praises the machine, the memory will be involved in the decision. So, in a similar context, the various Tokens may pass the activation value to either the punishment sign or to the reward sign. So, the machine's decision is a comprehensive statistics of all rewards and penalties, both may consider how to get rewards, and will consider how to avoid punishment, so the machine in choosing the response path, some local response path is the path to the reward, is the path to the punishment, so the machine needs to subdivide these path, to determine what is the path to the reward, what is the path to punishment. This segmentation process is to add more Tokens to this path, so as to form multiple segmentation paths (such as different scenarios, or different time points, or different factors, etc.), so that the machine can determine its response through the segmentation path, which is the core of segmentation imitation.

So, our machine does not need to accommodate the new "Fine tuning". It simply needs to achieve "Fine tuning" by accumulating memories. It can perform a "Fine tuning" at any depth, a "Fine tuning" in any field, and a superimposed "Fine tuning" in countless fields without "catastrophic forgetting". This is because it does not modify the past knowledge parameters, but rather does simply amplify the network.

In this case, the hypothesis is that during the day, the machine is lying (saving some power, getting rewards), after activating the punishment symbol of the owner's voice print, the machine needs to avoid the probability of the activated penalty symbol and increase the probability of the activated reward symbol. So, there are at least two Cases, 1, to reduce the probability of the "lying" concept and avoid punishment (such as being reprimanded); 2, to increase the probability of the "lying" concept and get a reward (such as saving power). At this time, the machine needs to make the best choice according to the principle of seeking advantages and avoiding disadvantages. At this time, the machine has to synthesize various response paths, and compare the statistical rewards and penalties.

If the machine is fully powered, the reward for saving power is small. After completing the chain-type associative activation process, only one penalty symbol achieved a high activation value. The machine will choose to avoid the punishment, because the highest reward value. So the machine, driven by profit maximization, will avoid punishment as a goal and start to build a response.

Assuming the machine is running low, the power savings is significant (assuming the machine has to lie down and charge). After completing the chain-type associative activation process, a penalty symbol obtains a high activation value, and a reward symbol also obtains a high activation value. According to the principle of seeking benefits and avoiding disadvantages, the machine simultaneously establishes two goals: to achieve rewards and avoid punishment. Because this is the highest reward value. So the machine, driven by profit maximization, will take the reward + avoid punishment as the goal and start to build a response.

Assuming the machine is fully powered, the machine now creates a level 2 goal: reducing the activation value of the "lying down" concept. Therefore, under the constraint of the level 2 objective, the machine looks for the propagation path to transfer the activation value to the "lying" concept, and creates the level 3 objective: reducing the activation value of the concepts on the propagation path. Thus, the machine finds that the main path to propagate the activation value to the concept of "lying" is the input of a set of self-state sensors. So the machine creates a level 3 target: reducing the probability of these input Tokens.

The machine will record various internal and external parameters of each training, using memory and forgetting mechanism, encourage it to imitate the reward parameters and avoid the reward parameters. In this way, an empirical connection is established between parameter combination, reward and internal and external environment. This is essentially a reinforcement learning process. Of course, humans can also imitate its form, implant innate knowledge (drive related) into the machine, or use the accumulated human experience to directly modify the knowledge of the machine so that it converges as soon as possible. So in different environments, the environment Tokens will

automatically activate the most relevant memories, by imitating these experiences, passing similar combinations of parameters to the machine's motor system (including parameter types and their temporal order, these processes are all automatically completed). This allows the machine to stand up in various environments, reducing the probability of "lying" related Tokens.

Assuming that the machine is then low, the machine's experience in achieving rewards will allow it to lie down, increasing the probability of charging related Tokens implementation. The experience of avoiding punishment, it imitates past experience and explains to the owner why he is doing so. The machine then creates a level 2 objective: raising the activation value achieved by the "charge" concept. To ic the past experience of avoiding "punishment". So the machine may create level 3 goal: "to explain the reason of their behavior", because the "Tokens combination" in memory, and "avoid punishment" Tokens combination there are close relationship between activation value transfer, so the goal of the machine is to improve and specific Tokens combination (to explain their behavior) occurrence probability. So the next level of decision of the Tokens combination is: the experience of language organization is activated.

This process proceeds iteratively, and each time a new reward and penalty symbol is activated. The machine counts the activation values of these reward and penalty symbols until the activation values of the reward and penalty symbols converge. Then the machine establishes the optimal response path.

The machine then goes into the imitation execution process. The decision path of the machine needs to be decomposed iteratively up to the underlying drive parameters, so that it can issue the drive command by imitating the parameter configuration in experience to imitate the execution.

In practice, experience and reality can always only be partially matched, so the generalization between experience and reality can only be realized by imitating their common Tokens combination mode.

Among these pathways, those composed of high activation value Tokens are the top-level imitation pathways. If the imitation path does not contain a direct underlying driver command combination, then more Tokens (lower activation value Tokens) is added in, and then the imitation path becomes a different combination of multisegment paths formed by more Tokens. This is what segmented imitation means.

That is to say, when we face a large path without the appropriate experience, we can then refine it and decompose it into several small response path segments. For each small path segment, we can look for the right experience to generalize the experience. If it is still impossible to decompose to the direct underlying driver command combination, then repeat the process by adding more Tokens, decompose the response path into more small path segments, and then find the right experience to generalize the experience. If you still can't break down to the direct underlying driver command combination, then repeat the process until it breaks down into the direct underlying driver command combination.

The above process continues iteratively. There may be new Tokens input all the time. Whenever a new Tokens input, the machine needs to do the chain association activation process again. After completion, the distribution of activation values in the memory bank changes, so the machine needs to restart the decision-making process. So in this process, the optimal decision of the machine may be to put down some of the current goals and start to pursue the latest goals.

So, our machine produces its own goals and can constantly change its own goals, so its decisions are very flexible and match the environment.

So in the example above, the possible result of the machine is to stand up immediately, improve the resolution of the sound processing system, and turn around to observe the owner's posture, movement and expression, but until this moment, the owner may have just said " Hello..."Word, the latter words have not yet begun.

So, our machine is human-like intelligence, and its understanding of information comes from its own experience, not from the statistical process. Only in this way can our machines have personalized services.

A thousand housewives, with a thousand different requirements. The artificial intelligence obtained through the knowledge statistics, the robot that cannot update the knowledge in real time, can never enter the home, and can never enter the hearts of the housewives. Their landing scenarios will be very limited, and our solution, which is the real general artificial intelligence, and it may change the face of the world.

9 Conclusion

We believe that the development of artificial intelligence can be approximately divided into different stages: (1) the "feature exploration" stage. Before deep learning, it was mainly focused on the "manual exploration" stage. After deep learning, focus on the "machine exploration" phase.(2) After the realization of real attention (Transformer), the machine realizes the "knowledge generalization" after the initial alignment of the machine's "knowledge coordinate base cluster"

and the human "knowledge coordinate base cluster (concept)". In the face of human tasks, machines can show certain intelligence through "knowledge generalization".[29][30]

The one-dimensional attention mechanism brings about a large model of language. Two-dimensional attention mechanism brings about image generalization. Three-dimensional attention mechanism, can achieve 3D creative ability. The four-dimensional (3 D + time) attention mechanism can realize the generalization of dynamic processes: it will bring video generation and robot services in limited scenes.

But we believe that only by increasing the "vitality: the fifth dimension, self-demand", can we bring the real "soul" to the machine intelligence. And the big model doomed it to achieve the "fifth dimension". And our solution can give "life" to the machine, so it can become a true "universal artificial intelligence".

So, we believe that AI needs to move on to the next stage: the "autonomous interaction" stage."Autonomy" means that the machine is no longer a silent "machine", it can spontaneously produce behavior (which is equivalent to programming itself), and the machine explores knowledge (for example, actively interacting with the environment to acquire knowledge)."Interaction" means that the machine can interact with the environment in real time, update its knowledge in real time, and make continuous decisions to complete complex tasks in an unfamiliar environment[29].

Many famous scholars have put forward their own views on how to move to the real general artificial intelligence, for example, Professor Lecun proposed the "world model", Professor Zhu Songchun also proposed the four characteristics of general artificial intelligence:

(1) can perform unlimited tasks;

(2) can independently generate new tasks;

(3) valuable system driven;

(4) has a world model reflecting the real world.

Obviously, our plan is a response to the ideas of Professor Lecun and Professor Zhu Songchun.

General artificial intelligence is the original intention of artificial intelligence, but also the crown of artificial intelligence. We present a set of technical solutions for implementing general AI, including implementation steps with Step by Step. In reference [25][26][27][28], we reveal in detail the technical steps to achieve this path by patent form. It may be the right path to lead humanity to general artificial intelligence.

References

[1] Reed S , Zolna K , Parisotto E , et al. A Generalist Agent[J]. 2022.

[2] Optimizing Language Models for Dialogue. `https://openai.casa/blog/chatgpt/`

[3] Ouyang L , Wu J , Jiang X , et al. Training language models to follow instructions with human feedback[J]. arXiv e-prints, 2022.

[4] Vinyals O , Ewalds T , Bartunov S , et al. StarCraft II: A New Challenge for Reinforcement Learning[J]. 2017.

[5] Quanjun Zhang,Chunrong Fang,Yuxiang Ma,Weisong Sun,Zhenyu Chen.A Survey of Learning-based Automated Program Repair. arXiv:2301.03270,2023

[6] Antonio Mastropaolo, Luca Pascarella, Emanuela Guglielmi, Matteo Ciniselli, Simone Scalabrino, Rocco Oliveto, Gabriele Bavota. On the Robustness of Code Generation Techniques: An Empirical Study on GitHub Copilot. arXiv:2302.00438, 2023

[7] Introducing ChatGPT. `https://openai.com/blog/chatgpt/`

[8] Chenglei Si, Zhe Gan, Zhengyuan Yang, Shuohang Wang, Jianfeng Wang, Jordan Boyd-Graber, Lijuan Wang. arXiv:2210.09150, 2023

[9] An experimental open-source attempt to make GPT-4 fully utonomous. `https://GitHub.com/Significant-Gravitas/Auto-GPT`

[10] Zhuosheng Zhang, Aston Zhang, Mu Li, Alex Smola. Automatic Chain of Thought Prompting in Large Language Models. arXiv:2210.03493, 2023

[11] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. In Thirty-sixth Conference on Neural Information Processing Systems (NeurIPS 2022), 2022a. `https://arxiv.org/abs/2201.11903`

[12] AUTO GPT,`https://auto-gpt.ai/`

[13] ERNIE Bot,https://mp.weixin.qq.com/s/0-8X9FPouteKzNiK6DPaiA

[14] Mccann B , Bradbury J , Xiong C , et al. Learned in Translation: Contextualized Word Vectors[J]. 2017.

[15] Vaswani A , Shazeer N , Parmar N , et al. Attention Is All You Need[J]. arXiv, 2017.

[16] Cheng J , Dong L , Lapata M . Long Short-Term Memory-Networks for Machine Reading[J]. 2016.

[17] Lin Z , Feng M , Santos C , et al. A Structured Self-attentive Sentence Embedding[J]. 2017.

[18] Parikh A , Tckstrm O , Das D , et al. A Decomposable Attention Model for Natural Language Inference[J]. 2016.

[19] Paulus R , Xiong C , Socher R . A Deep Reinforced Model for Abstractive Summarization[J]. 2017.

[20] Tamkin A , Brundage M , Clark J , et al. Understanding the Capabilities, Limitations, and Societal Impact of Large Language Models[J]. 2021.

[21] Kodge S , Roy K . BERMo: What can BERT learn from ELMo?[J]. 2021.

[22] Lee H , Xu G . Optimizing Policy via Deep Reinforcement Learning for Dialogue Management[C]// 2018 IEEE International Conference on Big Data and Smart Computing (BigComp). IEEE Computer Society, 2018.

[23] Ouyang L, Wu J,Jiang X,et al. Training language models to follow instructions with human feedback[J]. arXiv e-prints, 2022.

[24] Sunehag P, Hutter M. Principles of Solomonoff Induction and AIXI[C]// Ray Solomonoff memorial conference. Research School of Computer Science, Australian National University, Canberra, ACT, 0200, Australia;Research School of Computer Science, Australian National University, Canberra, ACT, 0200, Australia, Department of Computer Science, ETH Zurich, Switzerland, 2011.

[25] Yongcong Chen,Ting Zeng Xingyue Chen. Method for implementing universal artificial intelligence. CN111553467A[P]. 2020.

[26] Yongcong Chen,Ting Zeng,Xingyue Chen. A Machine Intelligence Implementation Method Imitating Human Intelligence, CN111582457A[P]. 2020.

[27] Yongcong Chen,Ting Zeng Xingyue Chen.Establishment of artificial general intelligence system, CN112016664A[P]. 2020.

[28] Chen Y , Zeng T , Chen X . ESTABLISHMENT OF GENERAL-PURPOSE ARTIFICIAL INTELLIGENCE SYSTEM:, US2022121961A1[P]. 2022.

[29] Yongcong Chen, Ting Zeng, Xiaoyi Qian. A Preliminary Study on the Capability Boundary of LLM and a New Implementation Approach for AGI. ChinaXiv:202305.00046

[30] Yongcong Chen, Ting Zeng, Jun zhang. A New Solution for Realizing True Artificial General Intelligence. ChinaXiv:202308.00026